NBS BUILDING SCIENCE SERIES 136

Organization of Building Standards: Systematic Techniques for Scope and Arrangement

U.S. DEPARTMENT OF COMMERCE • NATIONAL BUREAU OF STANDARDS

NATIONAL BUREAU OF STANDARDS

The National Bureau of Standards[1] was established by an act of Congress on March 3, 1901. The Bureau's overall goal is to strengthen and advance the Nation's science and technology and facilitate their effective application for public benefit. To this end, the Bureau conducts research and provides: (1) a basis for the Nation's physical measurement system, (2) scientific and technological services for industry and government, (3) a technical basis for equity in trade, and (4) technical services to promote public safety. The Bureau's technical work is performed by the National Measurement Laboratory, the National Engineering Laboratory, and the Institute for Computer Sciences and Technology.

THE NATIONAL MEASUREMENT LABORATORY provides the national system of physical and chemical and materials measurement; coordinates the system with measurement systems of other nations and furnishes essential services leading to accurate and uniform physical and chemical measurement throughout the Nation's scientific community, industry, and commerce; conducts materials research leading to improved methods of measurement, standards, and data on the properties of materials needed by industry, commerce, educational institutions, and Government; provides advisory and research services to other Government agencies; develops, produces, and distributes Standard Reference Materials; and provides calibration services. The Laboratory consists of the following centers:

Absolute Physical Quantities[2] — Radiation Research — Thermodynamics and Molecular Science — Analytical Chemistry — Materials Science.

THE NATIONAL ENGINEERING LABORATORY provides technology and technical services to the public and private sectors to address national needs and to solve national problems; conducts research in engineering and applied science in support of these efforts; builds and maintains competence in the necessary disciplines required to carry out this research and technical service; develops engineering data and measurement capabilities; provides engineering measurement traceability services; develops test methods and proposes engineering standards and code changes; develops and proposes new engineering practices; and develops and improves mechanisms to transfer results of its research to the ultimate user. The Laboratory consists of the following centers:

Applied Mathematics — Electronics and Electrical Engineering[2] — Mechanical Engineering and Process Technology[2] — Building Technology — Fire Research — Consumer Product Technology — Field Methods.

THE INSTITUTE FOR COMPUTER SCIENCES AND TECHNOLOGY conducts research and provides scientific and technical services to aid Federal agencies in the selection, acquisition, application, and use of computer technology to improve effectiveness and economy in Government operations in accordance with Public Law 89-306 (40 U.S.C. 759), relevant Executive Orders, and other directives; carries out this mission by managing the Federal Information Processing Standards Program, developing Federal ADP standards guidelines, and managing Federal participation in ADP voluntary standardization activities; provides scientific and technological advisory services and assistance to Federal agencies; and provides the technical foundation for computer-related policies of the Federal Government. The Institute consists of the following centers:

Programming Science and Technology — Computer Systems Engineering.

[1] Headquarters and Laboratories at Gaithersburg, MD, unless otherwise noted;
mailing address Washington, DC 20234.

[2] Some divisions within the center are located at Boulder, CO 80303.

NBS BUILDING SCIENCE SERIES 136

Organization of Building Standards: Systematic Techniques for Scope and Arrangement

James Robert Harris
Richard N. Wright

National Engineering Laboratory
National Bureau of Standards
Washington, DC 20234

U.S. DEPARTMENT OF COMMERCE, Malcolm Baldrige, Secretary
NATIONAL BUREAU OF STANDARDS, Ernest Ambler, Director

Issued September 1981

Library of Congress Catalog Card Number: 81-600124

National Bureau of Standards Building Science Series 136
Nat. Bur. Stand. (U.S.), Bldg. Sci. Ser. 136, 278 pages (Sept. 1981)
CODEN: BSSNBV

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 1981

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, DC 20402
Price **$7.00**
(Add 25 percent for other than U.S. mailing)

EXECUTIVE SUMMARY

An individual who refers to a standard, code, or specification should be able to find with ease and confidence the provisions needed. The organization of a standard determines whether provisions can be found reliably and efficiently. Organization has objective qualities that allow it to be treated formally. The organization of a standard deals with both the scope (the range of subject matter of the provisions) and the arrangement (the grouping and ordering) of the provisions it includes.

This report describes an innovative method for indexing and outlining that provides a systematic and effective tool for organizing standards or codes. Previously developed decision table and information network systems for analysis and representation of the provisions of a standard provide the context and need for an organizational system that capitalizes on their objective qualities. The development of the system is aided by the sciences of classification and linguistics.

The basic element is the development of a classification for the provisions of the standard. Development of the classification constitutes a formal treatment of scope. The classifiers are keys to naming chapters and sections. Development of an outline from the classifiers constitutes a formal treatment of the arrangement.

Necessary and desirable qualities for an organization are identified, verified, and adopted as objectives and guidelines. Two functional groups of provisions are identified: requirements and determinations. This grouping makes possible a clear definition of the interface of the organizational system with the decision table and information network systems and provides part of the basis for systematic classifecation.

A relevant basis is found for classifying requirements using an idealized model of the relation between syntax and semantics. This shows that a basic requirement names a thing as its subject and contains a required quality for that thing as its predicate. A faceted structure is recommended for a classification system that can meet the potentially conflicting demands of users with dissimilar purposes or backgrounds. The faceted classification provides a clear division between levels that are strictly logical and those that are not. Basic categories are expressed for engineering design standards for buildings to allow a meaningful starting point for their organization. Similar use of basic categories is recommended for other types of standards.

A technique based on the performance concept is developed for systematic formulation of new standards. It is applicable to standards that are not expressed in a performance oriented fashion.

Procedures are developed for forming an index and for forming outlines, and appropriate measures are defined for the comparison of alternate outlines for the same standard. Criteria for placement of provisions in outlines and for construction of outlines from the classification are proposed to promote the objectives of organization. Various outlining techniques are explored, and a computer algorithm for an interactive style of outline generation is developed and tested.

ABSTRACT

Standards should be organized so that they provide reliable and quick access to the provisions of the standard. Organization is considered to deal with both the scope and the arrangement of the provisions of a standard. It is found to have objective qualities that allow it to be treated formally. Necessary and desirable qualities for an organization are identified, verified, and adopted as objectives and guidelines. The basic element of the system for organizing standards is the classification of the provisions of a standard. A faceted structure, providing a clear division between those levels that are strictly logical and those that are not, is recommended for the classification system. A relevant basis is found for classifying requirements using an idealized model of the relation between syntax and semantics. Development of the classification constitutes a formal treatment of scope. The classification is easily transformed into an index. Development of an outline from the classifiers constitutes a formal treatment of the arrangement. Criteria for placement of provisions in outlines and for construction of outlines from the classification are proposed to promote the objectives of organization. A computer algorithm for interactive outline generation is developed and evaluated. Measures are defined for the comparison of alternate outlines for the same standard.

Key words: arrangement; building; classification; code; engineering; organization; provisions; scope; specification; standard; system analysis/engineering.

Cover: The large variety of buildings to which standards must apply requires sound principles for organizing standards.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

Facing page: The construction of a building requires the use of many standards.

CHAPTER 1

INTRODUCTION

1.1 Overview

Any individual who refers to a code or standard should be able to find with ease and confidence the provisions needed. Officials who judge compliance are reluctant to accept new technology in codes and standards [107],* in part because they cannot be confident of finding the correct provisions. Designers are reluctant to use unfamiliar codes and standards, often because it is hard for them to find and to be sure they have found all the relevant provisions [141]. The organization of a standard determines whether provisions can be found reliably and efficiently by the reader.

* The numbers in brackets correspond to the sources cited in REFERENCES.

This report describes an innovative method for indexing and outlining that provides a systematic and effective tool for organizing standards or codes. Organization deals with both scope and arrangement. Scope is the range of subject matter of the standard. Arrangement is the grouping and ordering of the provisions within the standard.

The method is a modification and extension of similar methods for outlining presented by Fenves, Wright, Nyman, and others [39, 51, 96, 97]. Elements of the method are founded in logic, classification theory, taxonomy, operations research, linguistics, and performance theory. The method is part of an overall system for the formulation, analysis, and expression of provisions in codes and standards. (The overall system is briefly described in chapter 2.)

The basic element is the development of a classification for the provisions of the standard. An outline of classifiers (keywords) is constructed from this classification, then converted to an outline of provisions. Development of the classification constitutes a formal treatment of scope, while development of an outline from the classifiers constitutes a formal treatment of the arrangement.

Provisions are classified according to their meaning (semantics) and structure (syntax). Explicit checks are made for clarity and completeness of a standard, and further checks and judgments regarding the consistency and correctness may be made. The use of performance related classifiers assures that the reason for each provision is understood, even if it is not expressed in the final text. The method is designed to provide reproducible results. The outline assures that the standard covers the scope and provides an unambiguous location for each provision, based on its subject matter, so clear access is promoted.

Writers of standards benefit by using the method because their task becomes more systematic, thereby possibly easier and more efficient. Readers, such as designers and compliance officials, benefit in the sense that they find it easier to locate pertinent provisions, and they have assurance that the standard is complete. The procedures encourage a consistent treatment of and explicit decisions about scope and arrangement by standards writers.

1.2 Objectives of Organization

The primary objectives of organizing the provisions within a standard or code are to define the scope and provide reliable and quick access to the provisions. The scope denotes the substance of the standard -- what entities must have what qualities in order to meet the purpose of the standard. An effective organizational system can assist standards writers in defining the scope.

Access is of paramount concern to successful use of a provision. Examination of a code or standard will show that there are five basic means for locating provisions within the text:

1) the table of contents
2) the index
3) headings that are printed within the text
4) proximity of related provisions
5) cross references written in the text

Each of these may make use of a labeling (numbering) system to identify sections of the text. While not every standard is organized with all of these means, each is important enough to merit consideration in planning the organization of a standard.

The first three means are essentially sets of titles, or headings. The systematic method for organization described here develops outlines of headings in a manner designed to assure that clear access is promoted through the first and third means. Because this method clusters related provisions, it also provides the fourth means listed above. The second, an index, may be developed directly from the classification. The fifth means, a cross reference, is an instruction at one point in the text to refer to another section, or heading. Cross references are not provided in the organizational method, but they are provided for by the information network, a related portion of the overall system that is briefly described in chapter 2 of this report. The interface of the system for cross reference with the organizational system is treated in this study.

There are several important qualities relating headings and their subordinate provisions that are associated with a good organization:

A. Qualities necessary for an effective organization:

1) <u>Relevant</u>: Each heading must be significantly related to its provisions; it must concisely express their scope.

2) <u>Meaningful</u>: The intended readers must perceive the heading as being relevant to the provision.

3) <u>Unique</u>: The headings must be distinct from one another to allow readers to access provisions unambiguously.

4) <u>Complete</u>: The total set of headings must cover the entire scope of the standard and nothing more.

5) <u>Graded</u>: The headings must show a regular gradation in scope through the levels.

B. Additional qualities desirable for an efficient organization:

1) <u>Progressive</u>: The headings at any level should be ordered in a pattern significant to the reader.

2) Intelligible: The depth (the number of levels in an organization) and breadth (the number of headings at one level) should not exceed the average span of immediate memory of the reader.

3) Minimal: The headings should be permuted so that the number of headings is the minimum for meaningful access.

4) Even: The organization should divide the provisions so that depth and breadth do not vary greatly from one part to another.

These qualities are objectives for the organization; the systematic method of organization is based on principles that are derived to provide these objectives. Since these qualities are the foundation of this study, their rationale is presented.

Of the utmost importance in assuring reliable access is that the heading must correctly identify the provisions it is associated with. Given that headings are a means for the identification of provisions, the most straightforward proof that headings must be relevant is in the contradictory question: Of what use is an irrelevant heading? (Proof by "reductio ad absurdum" [105].) An irrelevant heading is not only of no use in finding a provision, but it can actually prevent a user from finding a provision. Consider the following heading and provision paraphrased from a recent set of provisions for the seismic resistant design and construction of buildings [7]:

> COMPONENT DESIGN
>
> When the direct attachment method is used for components with (required) performance levels of S (superior) or G (good), the manufacturer shall certify that the components will not sustain damage if subjected to forces equivalent to those resulting from Formula 8-2.

The heading "Component Certification" would be much more relevant. A reader searching for a provision about certification would in all likelihood completely miss the provision with the misleading heading "Component Design."

Use of relevant words alone will not guarantee that the reader will make the correct association, however. It is also necessary that the reader easily understands what the headings mean; the words must be in his working vocabulary. As an example, the same set of provisions for seismic design makes use of the term "Moment Frame" in many of its headings. Practicing structural engineers form a large segment of the probable users of those provisions, and a great many of them have preconceived notions as to what words are used to classify frames. After study of the definition of "Moment Frame" offered by the authors of the provisions, the structural engineers are likely to use any or all of the following terms of their own: "Portal Frame," "Rigid Frame," or "Unbraced Frame." Adherence to the quality of meaningful headings would lead to the selection of a familiar term with the smallest chance of misunderstanding for use in a heading (for example, "Unbraced Frame"). New terms with restricted meanings are very useful in codes and standards, but they can be counterproductive when used as headings. The "semantic differential" may provide a tool to measure meaning [100], but it is not applied in this study.

Headings that are not unique are potential sources for two problems. First, there is the chance that some readers will use the provisions associated with one of a set of similar headings and fail to check the provisions associated with the other headings. Second, the ambiguity is likely to delay and frustrate some readers.

Completeness is necessary because provisions that don't have headings are difficult to locate, and headings that aren't associated with provisions are meaningless distractions. Consider the following example from the seismic design provisions [7] (emphasis added):

> GROUP III
>
> Seismic Hazard Exposure Group III shall be buildings having essential facilities which are necessary for post-earthquake recovery. Essential facilities, and designated systems contained therein, shall have the capacity to function during and immediately after an earthquake. Essential facilities are those which have been so designated by the cognizant jurisdiction . . .

The underlined provision is hidden in the midst of a definition and covered by a heading promising only a definition. It should have a heading of its own, such as "Group III Functional Requirement."

Systems of headings that do not show a gradation of scope with level are prone to a fault similar to non-unique headings: a subsection that covers subjects outside of the scope of its parent section inevitably will cause some readers to overlook pertinent provisions and create serious confusion in the minds of others.

Progressive ordering of headings allows the reader to predict where provisions will be placed, thus allowing him to make use of the fourth means listed earlier, location by proximity to similar provisions. Graded and Progressive are two qualities that are combined by many authors into a single quality, hierarchical. This single term is used on occasion later in this report because it is admirably concise, but the reader should recognize that two qualities with different rationales are included in it. Furthermore, as is discussed in chapter 3, the term is often used in an ambiguous way.

The remaining qualities, Intelligible, Minimal, and Even, contribute to efficiency in the use of headings. Intelligible requires sufficiently few headings at any one level so that the reader will recall how the heading relates to the others at the same level, and sufficiently few at more fundamental levels in the organization so that one can recall how the heading fits into the overall document. Minimal avoids superfluous headings.

Even suggests that comparable attention be given to comparable segments of the scope of the document.

It does not appear possible to formulate absolute rules to insure any of these objectives, because instances exist for each in which it must be sacrificed to preserve some other (for example, to sacrifice the quality Minimal to preserve the quality Complete, or to sacrifice the quality Even to preserve the quality Unique). Yet each is desirable enough to include in the list of objectives. Measures for several of these objectives are demonstrated in subsequent chapters.

All nine of these objectives gain in importance because of the typical mode of use of a code or standard. Rarely does an individual read through the entire document; more often the reader will use it in a reference mode, reading only those sections necessary to evaluate one particular question. Thus a burden of responsibility is placed on the headings, both individually and as a set (recall that this includes the table of contents and the index), for they must correctly direct the reader to pertinent provisions. This burden does not exist in documents that are read from the front cover to the back; the casual attention paid to headings in such documents is insufficient for a code or a standard.

1.3 The Role of Classification

Establishing a classification for the provisions provides a systematic approach to organization by headings. By classifying, one successively divides a set of provisions systematically so that access and arrangement can be promoted. A classification nearly automatically gives an index for access purposes, and it can be used to generate various outlines for felicitous arrangement and access. Furthermore, the scope of a set of provisions is represented by the classification used, so that formulating a classification is an important step in formulating a set of provisions.

It is demonstrated (subsequently) that because the scope of a provision is its most distintive feature, it is logical to focus on this when classing it. As a result, the classification developed for a standard concisely represents the scope of the standard. Classing provisions by scope is made systematic by observing that many ordinary provisions have a common underlying structure (this is discussed in detail in sections 3.3 and 4.1). It can be postulated from this observation that a simple but rigorous grammar could be developed for the language of mandatory provisions, although this step is not taken. What is significant here is that a basic structure for a general classification system for provisions can be inferred from this common underlying structure of provisions (and other information about building technology). Thus, the definition of the form of a provision leads to the structure of a classification system that in turn becomes a tool for working with the scope and arrangement of a standard and the access to individual provisions within the standard. The nature of the information used to classify provisions determines the detailed structure of the classification, consequently, the relations between provisions are expressed naturally in the products of the process of classifying.

Two general uses of the organizational method can be addressed: the organization and formulation of a new standard and the reorganization of an existing standard. Classification is applicable to both. For the former, the initial work is to develop the classification, then to develop an outline, or "Table of Intents," and finally to fit provisions into the outline. For the latter, a classification is developed for the existing provisions, then the outline is improved by using the classification. The former can be called the "top down" approach; the generation of the outline is controlled completely by the classification. The latter can be called the "bottom up" approach; the generation of the outline has a second level of control imposed in that empty headings (headings not associated with any provisions) are not generated unless wanted.

1.4 Important Definitions

> It is important to remember that language is man's only tool for conscious thought and that if you can't define it you can't think about it. Hence the precision of your thought is limited by the precision of your language. [116]

The vocabulary of codes, standards, and specifications is complex enough to warrant an effort to set forth explicitly what the following terms will mean in this report. All quoted definitions are taken from Webster's New International Dictionary, Second Edition [131].

A code may be: "any systematic body of law, especially one given statutory force," or: "any system of principles, rules, or regulations relating to one subject." Generally, it will not be necessary to distinguish between these two meanings in this report. Where the distinction is necessary, the term "legal code" will be used to indicate a statutory law.

A specification is:

> the designation of particulars; notation of limits . . . (Arch. & Engr.) the description of work to be done.

For the purposes of this document, it is convenient to distinguish between specifications for the construction of a particular project, which will be called construction specifications, specifications for the properties or manufacture of a particular product, which will be called product specifications, and specifications for the evaluation of engineering or architectural design, which will be called design specifications.

A standard is:

> that which is established by authority, custom, or general consent as a model or example; criterion; test; in general,

> a definite level, degree, material, character, quality or the like, viewed as that which is proper and adequate for a given purpose.

The meaning of the terms code, standard, and specification overlap significantly. It will be convenient for the reader to understand these overlaps and how, in certain fields of interest, the words are somewhat arbitrarily taken to have very distinct meanings.

In the field of building regulation, for example, code is generally taken to mean a legal code that is enforced by some governmental jurisdiction. Among building designers, however, code is frequently used in the broader sense of the second definition quoted from Webster for code.

To those in building regulation, the meaning of standard becomes limited significantly if it appears in conjunction with the words national or consensus. "National" implies that the people or organizations preparing the standard are truly representative of all parts of the Nation. "Concensus" implies that the process used to prepare and approve the standard rigorously assures that all dissent is heard and resolved. Generally the word consensus is not used unless the procedures have been formally approved by the American National Standards Institute. Thus the American Concrete Institute Standard 318, "Building Code Requirements for Reinforced Concrete" [16], is referred to as a national consensus standard by building officials, but is referred to as a code by designers.

The term specification is used to mean construction specification by many involved in construction, yet the "Specification for the Design, Fabrication, and Erection of Structural Steel for Buildings" [119], published by the American Institute of Steel Construction, is a national standard to building officials and a code to many designers.

For the purposes of this report, code, standard, and specification are essentially the same: a set of provisions providing principles, models, rules, limits, and particulars that are established by some authority for some purpose. <u>Standard will be the word used for this meaning</u>.

A standard is considered to consist of a set of provisions, where a provision is (still quoting Webster's [131]):

> that which is stipulated in advance; a previous agreement; a proviso; as the provisions of a contract; the statute has many provisions.

For the purpose of this report, all of the meaningful content of a standard is contained in its provisions, so that "the provisions of the standard" is actually synonymous with "the standard."

There are many different types of provisions found in standards, and it is very difficult to offer precise definitions of any type or a complete list of all types. It suits this study to define a few special types and to observe that the most general form of a single provision is a single sentence. Some sentences can contain more than one provision while some provisions may spread across more than one sentence, even widely separated sentences. Some provisions are not expressed in sentences, but as tables, graphs, or equations, and not every sentence in a standard is a provision because provisions must stipulate something. Those sentences and phrases used in standards to smooth the textual flow or comment on the stipulations are not considered provisions because they do not exercise mandatory control over the items or actions addressed by the standard. Thus in this study standards are divided into two parts, provisions and commentary. Only provisions are studied, as they are considered to carry all the significant information in the standard.

The first type that must be defined for clear understanding is a requirement. A requirement is:

> that which is required; specifically, a requisite or essential condition; a required quality, course, . . ., a necessity, need.

Thus, provisions establish requirements, and it is perfectly valid to say "the requirements of these provisions." It is also valid to say that a provision is a requirement. Requirements are normally characterized by the fact that their evaluation yields a value of "satisfied" or "violated." Following is an example of a requirement from the new seismic provisions [7]:

> PROTECTED ACCESS
>
> Buildings assigned to Seismic Hazard Exposure Group III shall be accessible during and after an earthquake. Where access is through another structure, that structure shall conform to the requirements for Group III. Where access is within 10 feet of side property lines, protection against potential falling hazards from the adjacent property shall be provided.

The requirement could be called the "Group III Access Requirement." Note that there are three provisions involved in the evaluation of this requirement, one general provision for all buildings and two other provisions for particular instances of unique hazards envisioned by the author of the requirement.

A criterion is:

> a standard of judging; a rule or a test by which facts, principles, opinions, and conduct are tried in forming a correct judgement respecting them.

Thus many provisions in a standard are criteria. Note that the word standard is used in the definition of criterion. Standard and criterion are nearly synonymous when standard is used in a singular sense, yet criterion adds the implication of a measurable test. In a collective sense, a standard may consist of many provisions, therefore, many criteria. The following example of a criterion is taken from the new seismic design provisions [7]:

> EXCEPTION: The nondestructive testing rate for an individual welder may be reduced to 25 percent with the concurrence of the person responsible for the structural design, provided the reject rate is demonstrated to be 5 percent or less of the welds tested for the welder.

The comparison of the welder's reject rate to the numerical value of 5 percent is the criterion. Criteria are frequently used in standards to provide measures for judging qualitative requirements, which would otherwise be difficult to evaluate. Note in the earlier example used to illustrate a requirement, no criterion was given to judge the compliance.

A definition is a: "limitation, setting of limits . . . making definite and clear." Many provisions in standards are simply definitions of words, terms, or symbols that have special meaning in the context of the standard. Although many standards have a special section reserved for definitions, they can be treated just as other provisions. Nearly all standards have definitions scattered throughout the text, whether or not they have special sections for definitions. All three of the following examples of definitions are taken from the new seismic design provisions [7]:

> 1.4.1 SEISMICITY INDEX AND DESIGN GROUND MOTIONS
>
> The design ground motions are defined in terms of Effective Peak Acceleration or Effective Peak Velocity-Related Acceleration, represented by coefficients A_a and A_v, respectively.
>
> 2.1 DEFINITIONS
>
> The following definitions provide the meaning of terms used in these provisions:
>
> ...
>
> APPROVAL is the written acceptance by the Regulatory Agency of documentation which establishes the qualifications of a material, system, component, procedure, or person to fulfill the requirements of these Provisions for the intended use."
>
> ...
>
> 2.2 SYMBOLS
>
> The following symbols and their definitions apply to these provisions:
>
> ...
>
> g = the acceleration due to gravity
>
> ...

1.5 Application Potential

Standards exist for a great number of purposes. The method of organization developed in this study, along with the conceptual model of standards of which it is a part, has been applied primarily to standards for the design of buildings. It is expected that the same methods and models can be applied beneficially to other kinds of standards in the future. To facilitate such application, the following brief sketch of building design standards is offered.

Building design standards are an important component of the building regulatory system in the United States. Legal codes for building regulation have a long history--at least back to 1700 B. C. when King Hammurabi promulgated his famous code [109]. The regulation of buildings is not one of the powers that the U.S. Constitution establishes for the Federal government, thus the authority is reserved for the State governments [18]. In most States, this authority traditionally has been delegated to or assumed by the local governments, although there has been a trend in the past few years towards State-wide codes for some aspects of building regulation [24]. Because there are so many local government units in the country, there are thousands of different legal codes within the U.S. The building regulatory system has often been the subject of study (see [2, 17, 18, 42, 101, 128] for a few examples), with calls for reform [17, 18, 89, 110] and even for a Federal pre-emption of state and local codes [2, 41].

The drafting of a complete legal code for building regulation is a task of such magnitude that it is beyond the resources of all but the largest states or cities. Most local governments adopt one of four model codes [11, 90, 121, 126] that are available [42]. Three of these model codes are produced by professional associations of local building officials. The other one has been produced by an association of insurance underwriters and may be produced in the future by an association of State building officials. Local governments frequently amend the model codes as they adopt them. Thus, the legal codes in two jurisdictions may be different even though they are based on the same model [128, 129]. Nonetheless, the model codes do introduce a significant amount of much needed uniformity into the legal code picture. In early 1978, the three associations of building officials discussed studying the possibility of producing a single model code, but the proposal was dropped later that year.

The model codes are standards that are not quite national in the range of application, because the associations of building officials that produce three of them are generally regional [99]. They are not consensus standards in the sense established in section 1.4 [70]. (The association of state building officials has plans to process their model code as a national consensus standard.) Each of the model codes incorporates by reference many national standards, many of which are consensus standards. These national standards are coordinated and produced by over 150 different groups [31], such as American National Standards Institute (ANSI), American Society for Testing and Materials (ASTM), American Concrete Institute (ACI), American Institute of Steel Construction (AISC), Brick Institute of America (BIA), National Forest Products Association (NFoPA), National Fire Protection Association (NFiPA), etc. Thus, legal codes frequently have provisions that are authored by individuals or groups far removed from the governing jurisdiction and its unique problems [60]. Likewise, national standards that are produced by voluntary associations frequently have application and impact far from the control of the original authors (see figure 1.1). For correct use in such an environment, standards must be organized so that relevant provisions are easy to find and understand.

The philosophy of control exerted by standards can be characterized most easily by defining the extremes: performance and prescription. Performance standards control buildings by defining in human terms the attributes that the finished building must provide [132]. Performance requirements are independent of the particular objects that make up a building or the schemes used to design and construct them. Prescriptive standards control buildings by specifying the objects and schemes used [132], thus implicitly assuring that the building will provide certain attributes. Prescriptive provisions are also commonly referred to as specification provisions or descriptive provisions. Example provisions dealing with similar situations which are near these extremes follow:

Performance	Prescriptive
An acceptable level of protection against structural failure under extreme load should be provided. [103]	(Wood) studs in exterior walls and interior bearing walls of buildings not more than two stories in height shall not be less than 2 inches by 4 inches in size. [126]
The fan shall move not less than 860 cubic feet of free air per minute and not less than 780 cubic feet of air per minute against a one-eighth inch static pressure. [8]	The fan shall be a direct drive, propeller type fan mounted on a ball bearing, totally enclosed, one-sixth horsepower motor supported by a rigid frame attached to the fan housing. The air inlet shall not be less than ten inches in diameter. [8]

Performance standards allow great flexibility, thus promoting innovation on the part of designers and builders. "Pure" performance provisions are often difficult and expensive to apply in practice, however, because they do not provide any means to measure levels of performance in lieu of full scale testing and experience. On the other hand, prescriptive standards are quite easy to apply because they provide precise measures to determine compliance. They sacrifice flexibility in the bargain, however, and stifle innovation. Thus the status quo is preserved without regard to any possible shortcomings or side effects.

Almost no standard is purely performance or prescriptive. Most are a judicious mix of provisions that fall between the extremes. The recent past has seen a move towards the performance type, spurred

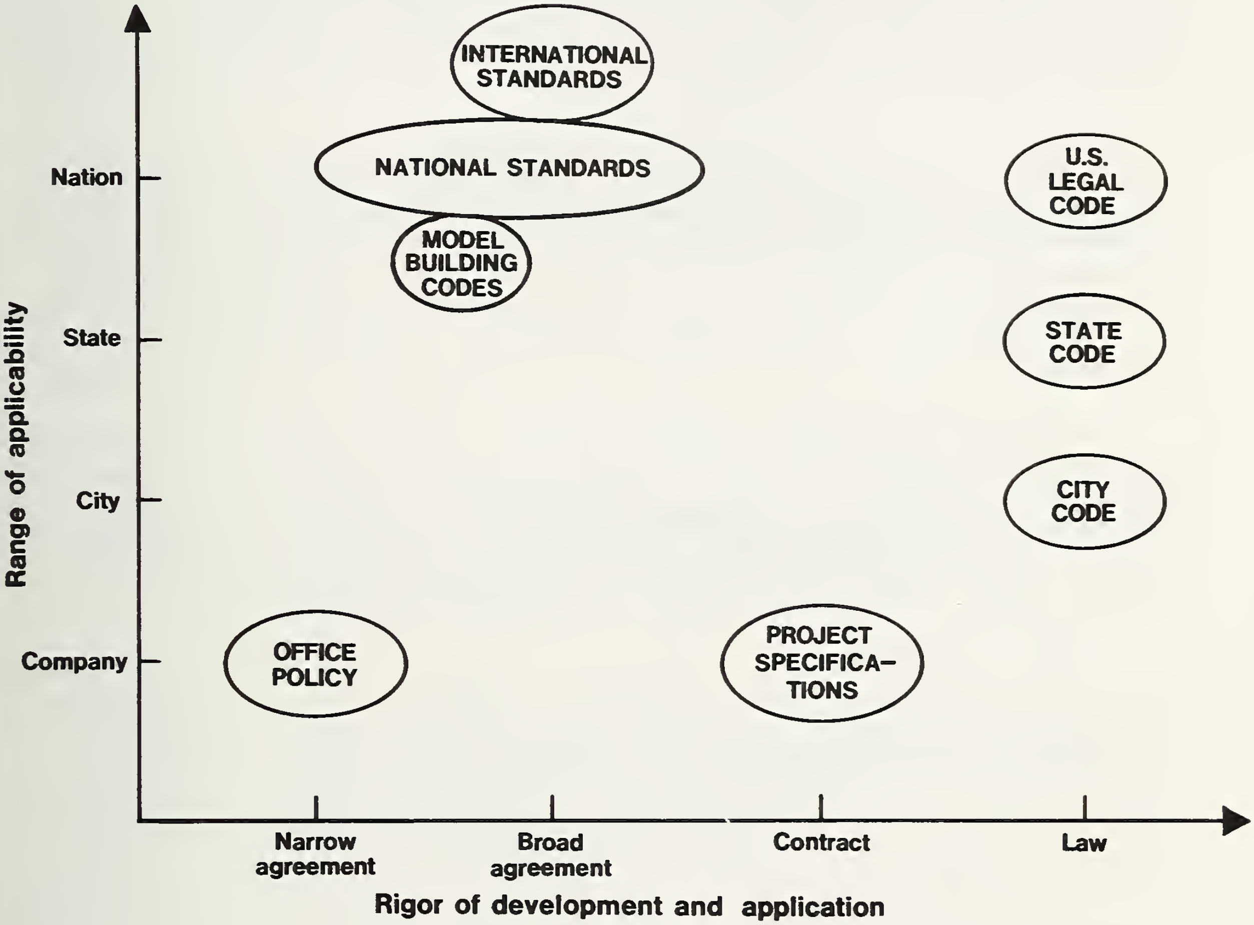


Figure 1.1 Range and Rigor of Standards
Note that many standards and model codes are developed at one level and subsequently applied through State and local adopting ordinances at much different levels.

on by "Operation Breakthrough," which provided guide performance standards for judging innovations in housing construction [74]. The problem of applying performance standards has been solved at least partially by following a format that requires the use of measurable provisions (called performance criteria) to judge each of the general provisions for attributes (called performance requirements) and the use of somewhat prescriptive provisions (called evaluation procedures) for the ways to measure the performance criteria [14]. (See figure 1.2 for a graphic representation of this format.) To be effective in allowing innovation, the performance criteria must be written in a manner to be independent of particular products and processes -- not an easy task.

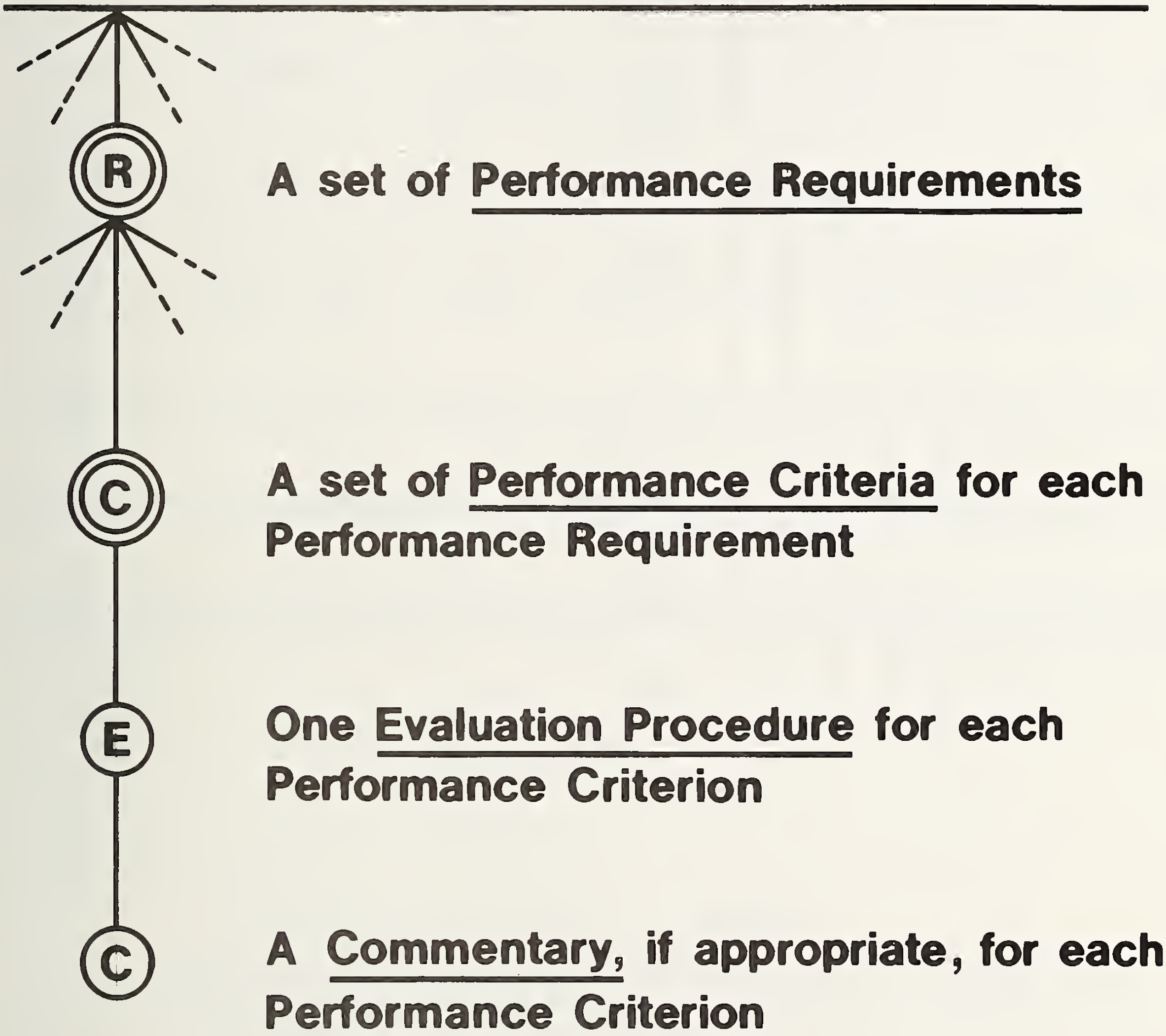
Although not called performance standards nor cast in the format just described, a class of standards for design of structures is somewhat similar in concept. Such standards contain provisions that are relatively independent of the type of structural material or assembly, thus not constraining the solution scheme too much. However, they contain complex, elaborate, and almost prescriptive provisions for the procedures of design and checking of such structures to assure that they will perform acceptably. The name procedural has been coined for this particular type of standard [38].

The method of organization developed in this study has been applied in studies of several building design standards over the range from performance to prescription. As described in later chapters, an understanding of the performance concept is vital to the successful organization of standards, even prescriptive ones. With this understanding, the method applies equally well to all types of building design standards.

Since building design standards are only one of many kinds of standards (material standards, testing standards, engineering practice standards, construction standards, quality control standards, etc.), a comment is in order on the applicability of the methods developed in this study (along with those described in chapter 2) to other types of standards. The objectives of organization (section 1.2), the principles of classification, the mechanics of producing an outline and an index, and the relation to other parts of the model of chapter 2 appear to be applicable to all standards (regulations, laws, or instructions) without much change. A small exception might be that an added objective of organization for construction project specifications is to separate the work so as to facilitate the taking of bids from various subcontractors [8, 123]. (Construction project specifications appear to be ahead of building codes in utilizing automatic data processing for generating new versions of provisions [48, 80]; the techniques reported here along with the techniques described in chapter 2 should aid such progress.)

However, the underlying structure of provisions and the basic classes that form the basis of a classification have been developed strictly within the context of building design and should be examined critically when applied in other fields. In particular, it would be expected that the basic classes might need to be different for different types of standards. With this caveat in mind, the application of the methods to other types of standards is encouraged. The benefits to be gained from clear organization seem well worth the effort.

PERFORMANCE STANDARD



A set of Performance Requirements

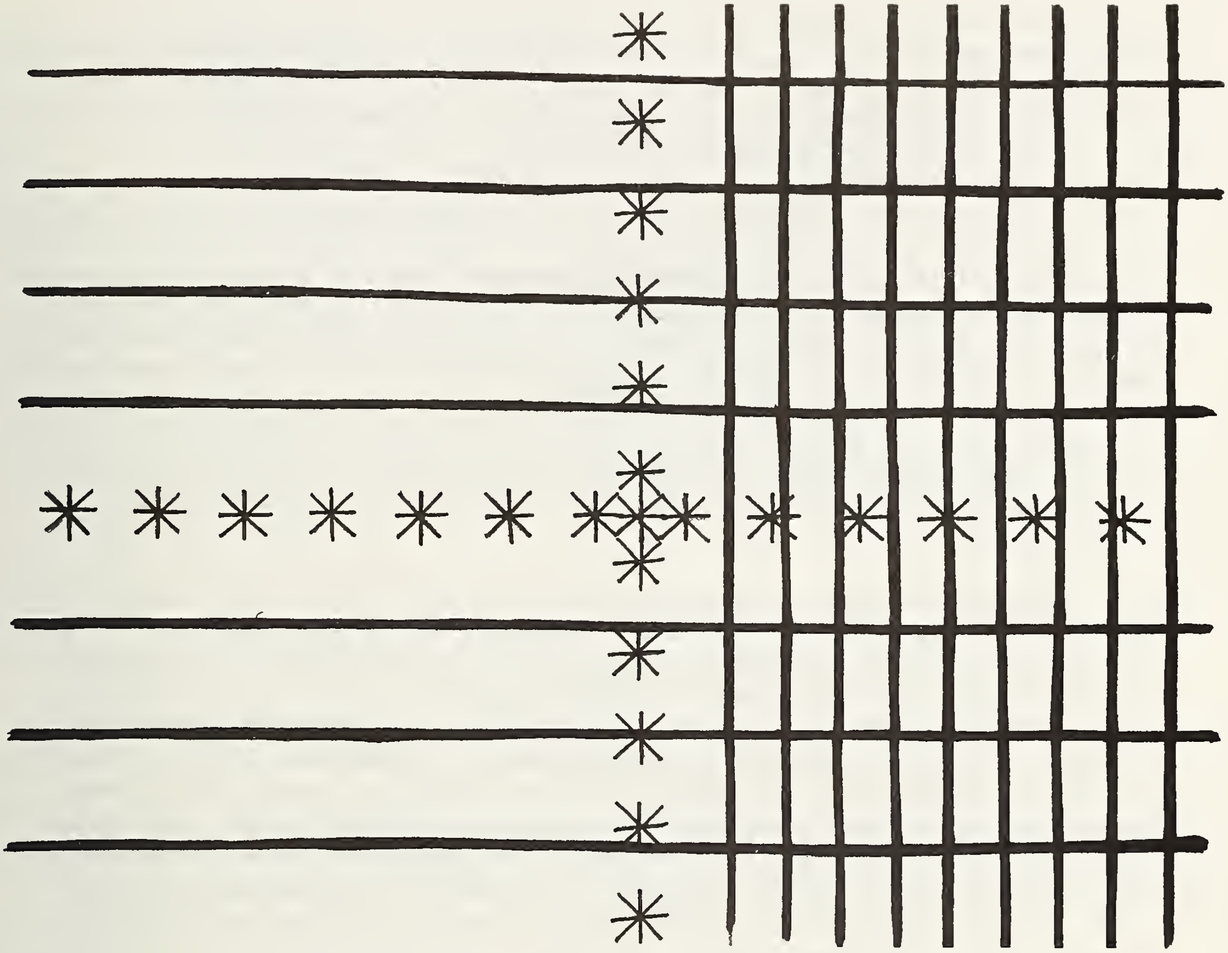
A set of Performance Criteria for each Performance Requirement

One Evaluation Procedure for each Performance Criterion

A Commentary, if appropriate, for each Performance Criterion

Figure 1.2 RCEC Format for Performance Standards

Facing page: The decision logic table is a key element in modeling standards.



CHAPTER 2

SYSTEM FOR THE MODELING OF STANDARDS

The conceptual model of a standard described here is a distillation of similar models that have been set forth in recent years. This chapter is a brief description of the state-of-the-art of analyzing and representing standards without consideration of the work of this study. It is included for background needed to understand the work of this study. If greater detail is needed, references 39 and 51 are most representative of the previous development that is summarized here. A prior call for the extension of the modeling into artificial intelligence [37] is not considered here.

2.1. Provisions, Datums, and Classifiers

As stated in section 1.4, all the meaningful content of a standard is contained in its provisions. The information contained in the provisions can be modeled as a set of units of information plus a set of relations among the units [39, 51, 96, 97]. Because the terminology used in describing the model is important, a few definitions are in order before examples are offered.* These units are used for two purposes within a standard:

1) to locate a particular provision
2) to evaluate another unit

It is possible for the same unit to serve both purposes. A unit used for the first purpose has been called an argument, a base, an element, a tag, and a descriptor in previous studies of this subject; such a unit is called a classifier in this report (the term argument will also be used for a special meaning, described later). A unit used to evaluate anothe unit (or ultimately to evaluate compliance with the standard) is called a datum. "Datums" and "data items" are used interchangeably for the plural, although in the light of Webster [131], the former ("something used as a basis for calculating or measuring") may be more appropriate than the latter ("something given or admitted esp. as a basis for reasoning or inference").

Excepting those provisions that are definitions, all provisions are modeled as giving a rule or rules for the evalution of a datum. (Although not stated as such, this is an implicit fact in most of the previous work.) Thus, the provisions quoted in section 1.4 under the heading "PROTECTED ACCESS" give rules for evaluating the datum representing the status of the requirement "Group III Access Requirement" (which may be either "satisfied" or "violated"). The criterion quoted in section 1.4 concerning a welder's reject rate is a part of a rule for evaluating the datum "Minimum Required Rate of Weld Testing for a Given Welder" (which may take the value of 25 percent, as stated in the quoted material, or 100 percent, as stated in prior material not shown in the referenced quotation). Examples of other typical datums are: the type of objects being addressed (building, floor, door, etc.); the measure of an object (area, height, etc.); and logical statements (fire extinguishing system present, wall separates buildings with different heights, etc.).

Datums are frequently distinguished by the type of value carried:

- numerical values (such as area)

- arbitrary values from a restricted set (such as type of object)

- the binary logical values of true and false--or equivalently, satisfied and violated. These are also termed Boolean values.

Datums are also distinguished by one further characteristic, their place in the precedence of evaluation:

- Input datums are those datums for which the standard contains no rules for evaluation. They are supplied by the user.

* A glossary of the special terms used for the description of the conceptual model is included following the references.

- Derived datums are those datums for which the standard provides rules for evaluation.

- Terminal datums are those derived datums that are not used within the standard for the evaluation of another datum. Each of them is the status of a requirement, although not all requirements are terminal datums.

In any standard, the names of the datums and the classifiers will not include all of the words used in the text. Two general classes of words used in the expression of standards are not captured in the datums and classifiers of the model: those words serving a syntactic function to indicate relations between units or unite the concepts into readable prose and those words used to comment on the meaning of the provisions. It is common for standards to contain commentary mixed in with provisions, although this may not be apparent without performing a thorough study.

Recent studies [25, 50, 124] have explored the fundamental element of the overall model, the datum. Building on the work of Fenves, Rankin, and Tejuja [38], Tavis and Melin [124] have proposed rigorous techniques for the translation from conventional textual expression of standards to the datums and their relations. These translation procedures are valuable additions to the analytical tools, providing as they do for the explicit recognition of the equivalence of different names for the same datum (apparently a common problem in standards-like documents, see [28, 29] for example), the ambiguity of different meanings implied by the same name in different contexts, and the assumptions mandated by implied but unnamed datums.

The relations between datums are modeled with decision tables and information networks, discussed in the following two sections. The relations among classifiers and between classifiers and datums are modeled in the "organizational system" and are discussed in section 2.4.

2.2 Decision Table

Herman McDaniel, an authority on decision tables, has written [82]: "It is not enough to write so that you can be understood. You must write so that you cannot be misunderstood." Decision tables greatly facilitate that goal. The decision table is used to represent and analyze the functional and logical relationships (that is, the rules) that establish the value for each derived datum. It is a way of dealing with the meaning of an individual provision. A decision table is simply an orderly presentation of the reasoning controlling any set of decisions. It is easily analyzed to assure that the reasoning process will always lead to a unique result and that no possibility exists for encountering a situation not defined. Another advantage of decision tables is that they readily describe situations involving parallel thought processes whereas written text and, to some extent, flow charts more readily describe a sequential thought pattern.

The following introduction to decision tables is brief. More detail can be found in the references [51, 81, 82, 104]. A decision table is composed of conditions, actions, and rules, arranged as shown in figure 2.1. A condition is a logical statement that can have only one of two values: true or false. An action in a general sense is any operation; for example, it may be the assignment of a value to a variable by means of a formula or a statement of control that may indicate the next procedure to be initiated. In the context of this model, an action assigns a value to one datum. A rule is a statement that prescribes a set of conditions in order that a specified set of actions can be performed.

Table 2.1 shows an example of a very simple decision table which represents the following provision from the Uniform Building Code (UBC) [126]: "No single floor area shall exceed that permitted for one story . . ."

The only condition is that the actual story area be less than the total allowable story area, and the two possible actions are that the requirement is either satisfied or violated. In the rules of the table, Y stands for true, N for false, and X for "take this action." (T and F are often used in lieu of Y and N.) The table is read rule by rule, "If the actual story area is less than the total allowable story area, then the single story area check is satisfied; if it is not, then the check is violated."

A decision table is essentially a structure for defining a set of related rules. Each rule contains an entry (value) for each of the conditions and an entry to indicate which action is to be executed for the rule. The rule can be thought of as a logical AND function, that is, the rule is not satisfied unless each of the condition entries it contains is matched.

Decision tables were developed during the late 1950's to describe complex logical problems for computer programming that had resisted description by flow charts and narratives. The decision table is obviously related to the truth table, a tabular method of testing the validity of propositional statements in formal logic [105], and it probably was originally derived from truth tables or similar logical devices. Pollack [104] and McDaniel [81, 82] describe the theory and practical use of decision tables as they had developed through the late 1960's, and Metzner and Barnes [85] includes some more recent information; Pollack is the most complete source.

Fenves [33] proposed the use of decision tables for the representation of engineering design provisions, citing the positive benefit of the capacit for formal analysis of completeness and unambiguity that is possible with them. In 1969 a significant portion of a standard for the design of steel buildings [119] was represented as a set of decision tables [36]. Other building standards have been represented as a set of decision tables since that time [3, 25, 50, 95, 98, 113].

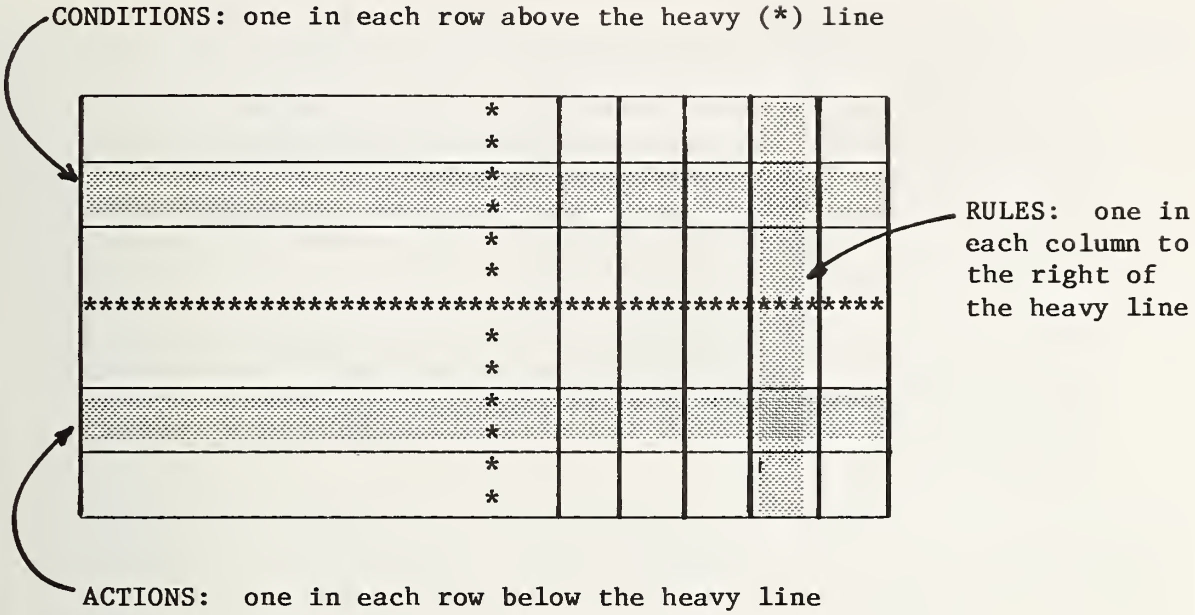

CONDITIONS: one in each row above the heavy (*) line
RULES: one in each column to the right of the heavy line
ACTIONS: one in each row below the heavy line

Figure 2.1 Conventional Structure of a Decision Table

Table 2.1 Decision Table for Single Story Area Requirement

		Rule 1	Rule 2
Condition 1	Actual story area ≦ Total allowable story area	Y	N
Action 1	Single story area requirement = satisfied	X	
Action 2	Single story area requirement = violated		X

The decision table of table 2.2 represents the following provisions for special occupancy situations in section 506(b) of the UBC [126]:

> Unlimited Area. The area of any one- or two-story building of Group F, Group G and Division 5 of Group E Occupancies shall not be limited, if the building is provided with an approved automatic fire-extinguishing system throughout, as specified in Chapter 38, and entirely surrounded by public space, streets or yards not less than 60 feet in width.
>
> The area of a Group G Occupancy in a one-story Type II, Type III Heavy Timber, Type III one-hour or Type IV building shall not be limited if the building is entirely surrounded and adjoined by public space, streets or yards not less than 60 feet in width.

It illustrates several more features of decision tables: 1) a single condition can be made up of several logical comparisons, 2) decision tables can have many conditions, 3) the condition entries in the rules can be other than Y or N, and 4) a special type of rule, called ELSE rule, may be used to collectively indicate all possibilities not explicitly expressed (the column under the E). Note that a condition containing a series of logical subunits connected by an and (a logical and) is true only if all of the subunits are true, whereas a condition containing a series of subunits connected by an or is true if any of the subunits are true.

The condition entry "•" (which does not occur in table 2.2) stands for immaterial, meaning that either a true or false value for that condition is acceptable for the rule with the "•". The significance is that the condition has no bearing on the rule and consequently need not be checked to verify that rule. The condition entry "-" means implicitly false, or false without testing. It is used to note that the value of that condition is predetermined by the value of some other condition expressed for that rule [104]. Since conditions two and three of table 2.2 contain opposing statements about the presence of a fire extinguishing system, they cannot both be true in the same rule. It is also possible to use the symbol "+" meaning implicitly true, but it does not occur in this table. The significance of implicit entries is that they show relations among conditions where such relations exist and they reduce the amount of checking necessary to verify a rule.

A great deal is learned about a set of provisions by formulating the decision tables for them. A singular advantage of decision tables, however, is that they lend themselves to a systematic analysis for completeness and uniqueness: complete in the sense that every possible set of values for the conditions will match some rule in the table, and unique in the sense that each possible set of values for the conditions will match one and only one rule. The most convenient way to

Table 2.2 Decision Table for Allowable Increases for Special Occupancy Situations

		Rules		
		1	2	E
Condition 1	All sides have separation and Minimum width of separation ≥ 60'	Y	Y	
Condition 2	Number stories ≤ 2 and Occupancy = F, G, or E5 and Fire extinguishing system present = true	Y	-	
Condition 3	Number of stories = one and Occupancy = G and Fire extinguishing system present = false and Construction type = II, III-H.T., III-1 hr., or IV	-	Y	
Action 1	Allowable increase (506 (b)) = unlimited	X	X	
Action 2	Allowable increase (506 (b)) = 0			X

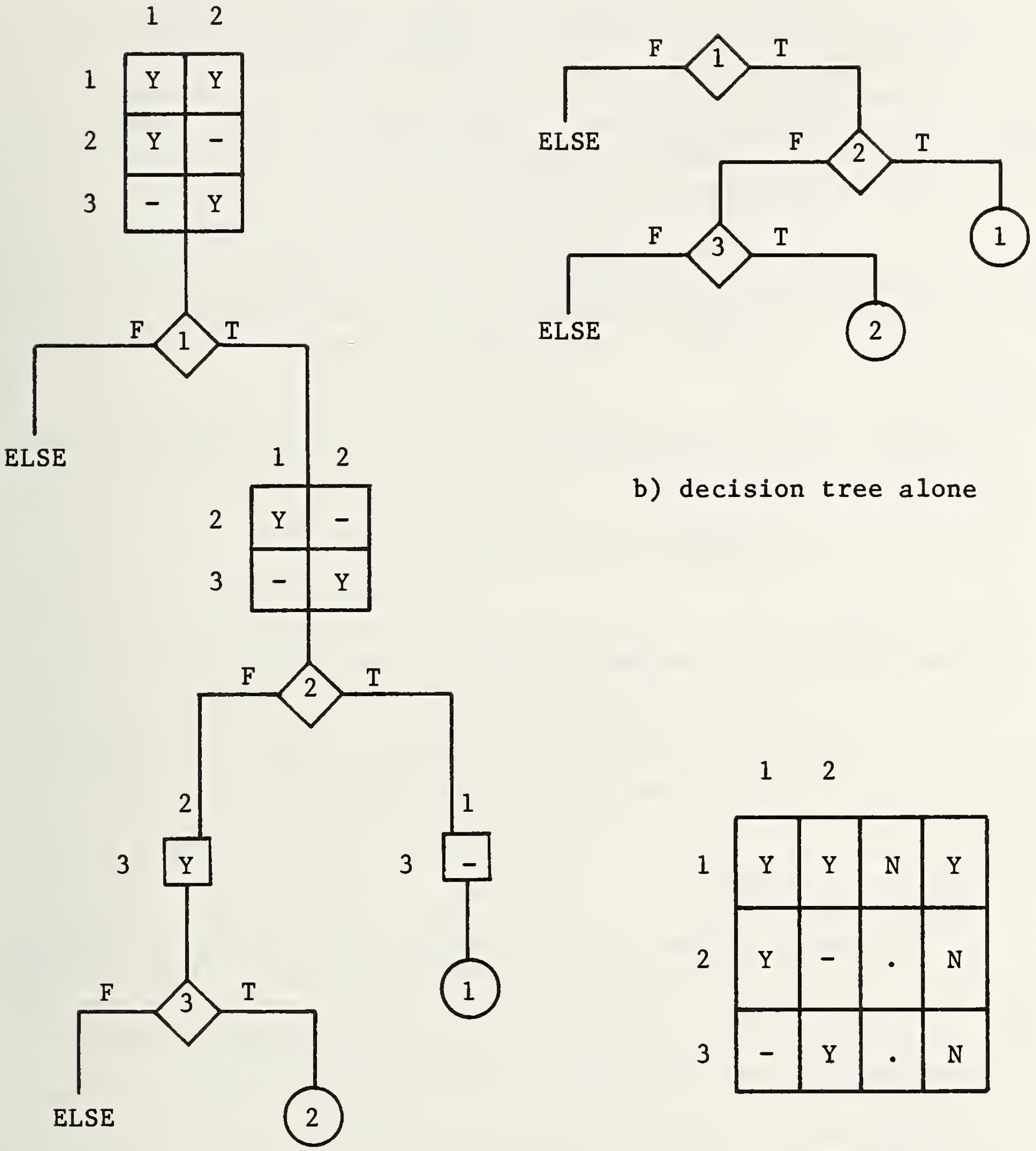
carry this analysis out is to construct a decision tree from the decision table. The upper portion of the rules (generally called the condition entry) from table 2.2 is shown in figure 2.2, along with the decision tree derived from it.

The decision tree is constructed by dividing the table into two subtables each time a condition is tested, one subtable containing those rules for which the condition is true, the other containing those rules for which the condition is false. A rule is isolated when all of its entries that are not either immaterial or implicit have been tested. Dividing table 2.2 by testing condition 1 results in two new subtables, one with rules 1 and 2 for the "true" branch and the ELSE rule for the "false" branch; since no rule contains an "N" or a "•" for condition 1, there are no rules in the subtable on the "false" branch. Testing condition 2 in the first subtable separates rule 1 from rule 2 and since rule 1 contains no other explicit entries, it is isolated from the table and shown as a circle terminating a branch on the decision tree. Testing condition 3 in the remaining subtable yields rule 2 and another ELSE rule. This decision tree shows that the decision table is incomplete because two ELSE rules exist and that the table is unique because no test of a condition yields more than one rule without conditions remaining to be tested. It is appropriate to continue to represent the table with an "E" for the ELSE rule as long as the same action is correct for all possible variations of the ELSE rule (the two possibilities are shown in part c of figure 2.2). The real power in this analysis is in detecting unforeseen ELSE rules.

The formal analysis of decision tables for completeness was explored in several studies by Noland, Feng, and others [92, 93, 94]. Each of these approaches dealt with proper ways to count the number of possible rules in tables with related conditions. Studies at the University of Illinois [51, 52, and 135] showed that the incorporation of implicit entries for related conditions, as described by Pollack [104], allowed the decision tree to be used for analysis of completeness and produced a computer program for this purpose.

Many algorithms for developing decision trees exist, particularly for the problem of deciding what order to use in testing the conditions, see [51] for examples. A particularly important contribution was that of Montalbano [88], who described two general algorithms for generating decision trees from a decision table, a quick rule that minimizes the total number of branches and a delayed rule that minimizes the average branch length. References 111 and 112 describe more sophisticated techniques. Computer software is available to generate decision trees from decision tables for the purpose of checking completeness and uniqueness [52, 135] but it is easy to do by hand for all but the largest tables once one has practiced the procedure.

Decision tables are useful tools in standards work for reasons beyond their capacity to offer an analysis for completeness and uniqueness of logic. They give an explicit and precise way to express provisions, albeit somewhat more space consuming than conventional prose. Also, decision tables



a) decision tree showing splitting of table into subtables

b) decision tree alone

	1	2		
1	Y	Y	N	Y
2	Y	-	.	N
3	-	Y	.	N

c) ELSE rules shown with original rules in tabular form

Figure 2.2 Example Decision Tree

can be used to facilitate the preparation of computer programs for checking compliance with standards. Software exists that practically make a decision table self-programming [33, 104]. Because decision tables are modular, future changes in standards are easier to incorporate into such computer programs; wholesale reprogramming is not necessary [33]. The power of decision table expression of a standard will become even more apparent in the following sections.

It also should be noted that not all rules for the evaluation of datums are conditional. There are instances in which a provision establishes one and only one way of evaluating a datum that does not depend on any special circumstance. The following example is from the new seismic design provisions [7]:

> The Occupancy Potential, OP, shall be determined in accordance with the following formula:
>
> $$OP = \frac{\text{Total area of all floors}}{\text{SFPO}} \qquad (13\text{-}1)$$
>
> where SFPO = the square feet per occupant for each floor as given in Table 13-A or as established by the cognizant jurisdiction.

If a decision table were to be written for OP, it would have no conditions and only one action (the formula given). Since no decision making is necessary, the name function has been used to describe such provisions.

2.3 Information Network

The information network is a directed graph used for a clear expression of the precedence relations between provisions. Each datum is a point, or node, on the network. The nodes are connected to their ingredient nodes by branches that represent the flow of information through a set of provisions from the input datums to the terminal datums.

The early case studies [36] that cast design standards in decision table form typically used three kinds of decision tables that were linked together in a network. "Switching" tables brought the reader to the applicable tables much as a table of contents would (shown clearly in [47]). "Testing" tables represented the most basic provisions (requirements) and typically carried a value of "satisfied" or "violated." "Working" tables were used to develop the data required for the testing tables. This network of decision tables was used in computer programs for the automated checking of design constraints (constraint processing) [45, 46, 93].

The network of decision tables became the current information network and became much more straightforward and usable when the action set of a decision table was limited such that only one datum would be evaluated by any decision table [135]. This allows a one-to-one correspondence between a network of decision tables and a network of datums, and as a by-product leads to decision tables that are generally small and simple.

The ingredients of a datum are defined as all those nodes that may be required for direct evaluation of the datum. All derived datums, whether decision tables or functions, have ingredients. For example, the ingredients of the "Single Story Area Requirement," shown in table 2.1, are the "Actual Story Area" and the "Total Allowable Story Area." Likewise, the ingredients of "Occupancy Potential," the function cited earlier, are the "Total Area" of All Floors" and the "Square Feet per Occupant." Figure 2.3 shows a small information network that connects ingredients to the datum given by the decision table of table 2.2.

The entire network for a standard can be assembled once each node and its ingredients are known. The assembly is generally performed on a computer because the manipulations are tedious, but the concept is relatively simple (a computer program for this is described in references 52 and 135). Figure 2.4 shows a larger portion of the information network developed from the same section of the UBC [126]. The portion enclosed by the dashed line is the same as figure 2.3. As before, the arrows point from the ingredient node. Any derived node (derived datum) can be said to be a dependent of each of its ingredient nodes. The assembled network shows that several nodes have multiple dependents, meaning that the node is used as an ingredient in more than one derived node.

The assembled network can be used to determine several other items of interest. The nodes that have no ingredients are input datums; their value must be supplied externally. The nodes that have no dependents are the terminal datums, also called output nodes. Each node can be assigned a level from input or output by counting the number of branches between the node and an input or output node. Where more than one path exists, the convention has been to use the longest path.

Two important items for any node are its global ingredience and its global dependence. The global ingredience of a particular node is the portion of the overall network that is located on branches pointing towards the node. Stating it another way, it is the subnetwork that begins at the node in question, then goes to each of its ingredients, which in turn are followed by each of their ingredients in a recursive manner until all the branches end at input nodes. The direction of traversing this network is counter to the direction of the branches.

The global dependence is constructed in a similar manner, except dependents are used, the final nodes are output nodes, and the direction of traversing is with the direction of the branches. The global ingredience shows all of the nodes that may have a direct or indirect effect on the node in question. The global dependence shows all the nodes that may be affected by the node in question. Referring again to figure 2.4, the global ingredience of node 8 would include all of the nodes shown except nodes 6 and 7. The global dependence of node 44 would include the nodes 7, 6, 40, 8, and 46.

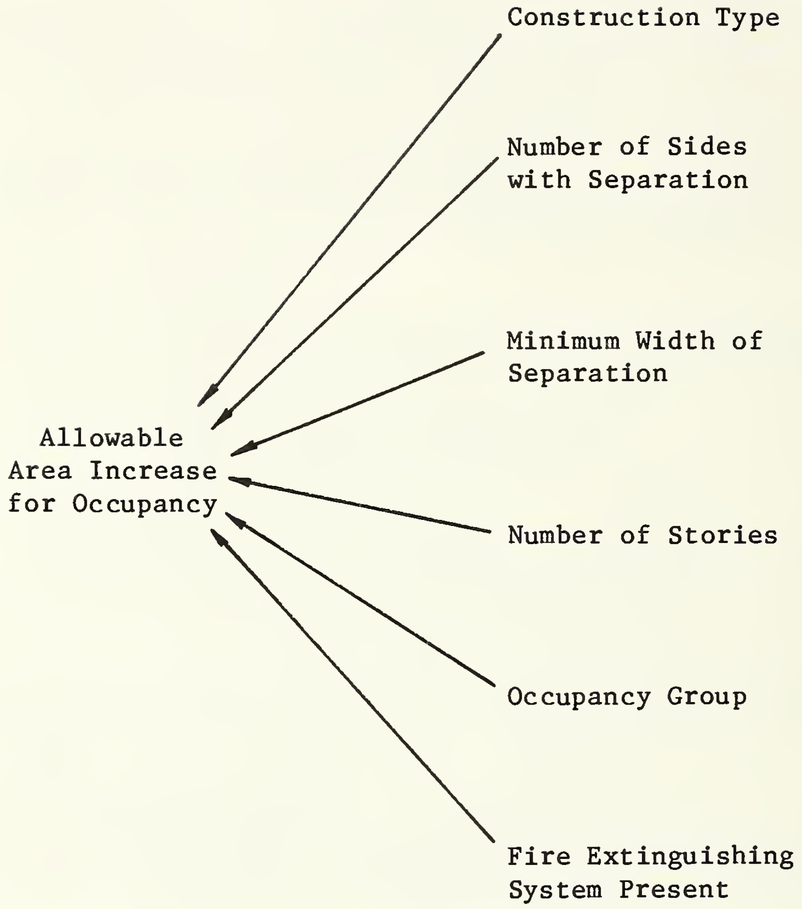

Construction Type
Number of Sides with Separation
Minimum Width of Separation
Allowable Area Increase for Occupancy
Number of Stories
Occupancy Group
Fire Extinguishing System Present

Figure 2.3 One Level of an Information Network

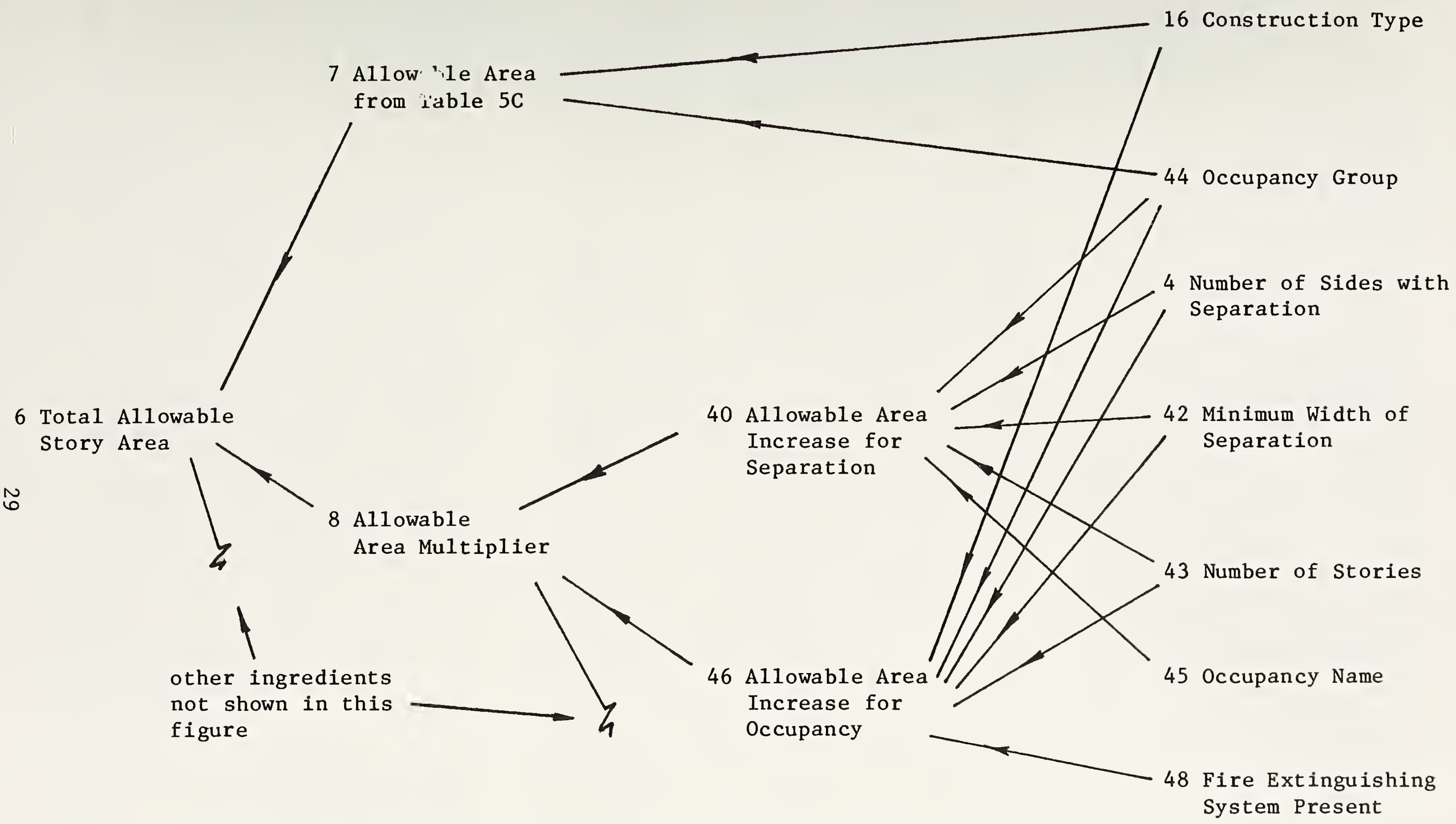

16 Construction Type
7 Allowable Area from Table 5C
44 Occupancy Group
4 Number of Sides with Separation
6 Total Allowable Story Area
40 Allowable Area Increase for Separation
42 Minimum Width of Separation
8 Allowable Area Multiplier
43 Number of Stories
46 Allowable Area Increase for Occupancy
45 Occupancy Name
other ingredients not shown in this figure
48 Fire Extinguishing System Present

Figure 2.4 Portion of an Information Network

The network of figure 2.4 had several closed meshes. Figure 2.5 shows the network of figure 2.4 with several branches replaced by dashed lines in order to form a spanning tree. (A tree is a graph with one root and no closed meshes, a spanning tree includes all the nodes in the original network, but deletes enough branches to avoid closed meshes [12].) The spanning tree representation of the information network is quite convenient. It allows computer driven representation of an information network. Figure 2.6 shows a spanning tree printed by a computer in an indented outline format. It is for the provisions limiting the area of a building from the UBC [126], from which several of the examples cited earlier have been taken. (The study that developed this network is reported in reference 49.) Note that the network of figure 2.5 is a small portion of the network of figure 2.6.

Each node in the network is printed in the column corresponding to its level from output. Printouts representing global dependence can also be generated, and for such printouts the columns used correspond to the level from input. The dotted lines represent the branches and the direction of the branches is to the left for ingredience networks (as figure 2.6 is) and to the right for dependence networks. The branches that were removed to make the spanning tree can be found by locating each occurrence of a node with a dash printed in front of it. The dash means that this node has already been printed in the spanning tree. To find the original occurrence of the node, progress up the display, remaining in the same column, until the node is printed without a dash. The original network had a branch from the dependent of the node with a dash to the node without the dash.

The formal analysis of the information network for cyclic loops and disconnected nodes is described in references 39 and 51. In addition, the latter study describes qualitative analyses of the information network for parallel dependence of similar provisions. Recent work [124] has extended the information network to represent equivalent and assumed names for datums.

The spanning tree display of an information network is particularly useful because it can be related to the order of narrative expression of a set of provisions. The global ingredience of a datum can be thought of as the road map for the complete definition of that datum, and there are two useful ways to traverse the road map for the purpose of expressing that definition. The first is called conditional ordering, or top down. As shown in figure 2.7, conditional ordering begins at the output node of the ingredience network, then proceeds along one path to an input node, always defining a node before defining its first ingredient [67]. The choice of which ingredient to take first is arbitrary in this model. When an input node is reached, the order reverts to the first dependent that has yet undefined ingredients, continuing this pattern until all nodes are defined. The second way to traverse the same network is called direct ordering, or bottom up. As shown in figure 2.8, the direct order begins at an input node and proceeds to define all the ingredients of a node before defining a dependent [67]. The choice of which input node to take first is, once again, arbitrary.

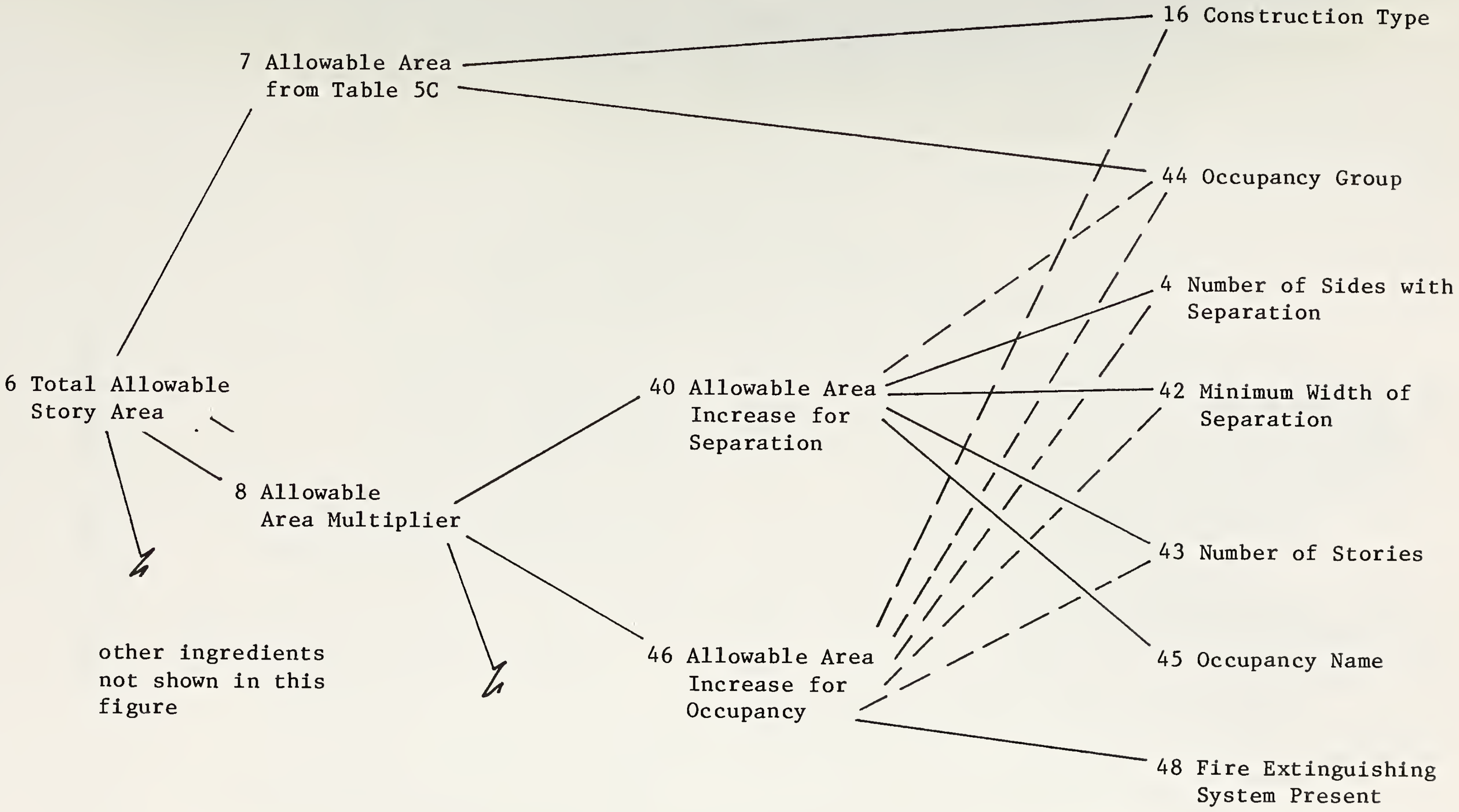

16 Construction Type
7 Allowable Area from Table 5C
44 Occupancy Group
4 Number of Sides with Separation
6 Total Allowable Story Area
40 Allowable Area Increase for Separation
42 Minimum Width of Separation
8 Allowable Area Multiplier
43 Number of Stories
other ingredients not shown in this figure
46 Allowable Area Increase for Occupancy
45 Occupancy Name
48 Fire Extinguishing System Present

Figure 2.5 Information Network Converted to a Spanning Tree

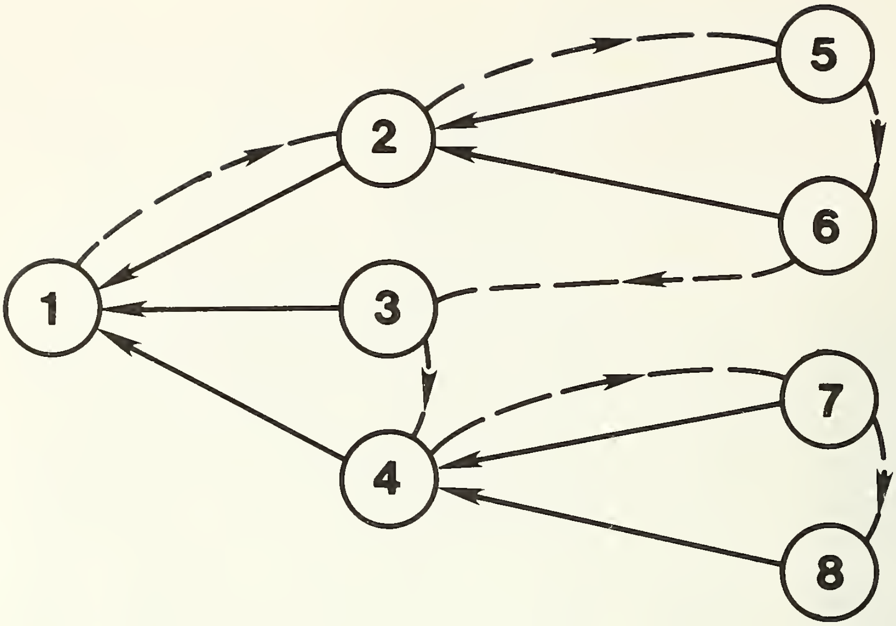
Note that the dashed branches are removed from the network of figure 2.4 to form a spanning tree

```
GLOBAL INGREDIENCE OF COMPLETE NETWORK

UNSORTED

EXTREME LEVEL FROM OUTPUT

   0         1         2         3         4         5         6         7         8
   1  AREA CHECK
   :.........2  SINGLE STORY AREA CHECK
   :         :.........4  ACTUAL AREA OF STORY
   :         :         :......................................14  AREA SEPARATION WALL CHECK
   :         :                                                 :........15  AREA SEP WALL FIRE RESISTANCE CHECK
   :         :                                                 :         :........16  CONSTRUCTION TYPE
   :         :                                                 :         :........17  FIRE RESISTANCE OF AREA SEP WALL
   :         :                                                 :         :........18  FIRE RESISTANCE OF OPENINGS IN AREA SEP WALL
   :         :                                                 :         :........44  OCCUPANCY GROUP
   :         :                                                 :........19  TOTAL WIDTH OF OPENINGS IN AREA SEP WALL IN STORY
   :         :                                                 :........20  TOTAL LENGTH OF AREA SEP WALL IN STORY
   :         :                                                 :........21  AREA SEP WALL EXTERIOR TERMINATION CHECK
   :         :                                                 :         :........22  A S WALL EXTENDS TO EDGE OF HORIZ PROJ ELTS
   :         :                                                 :         :........23  FIRE RES OF EXT WALL AT AREA SEP WALL
   :         :                                                 :         :........24  WIDTH OF FIRE RES EXT WALL AT AREA SEP WALL
   :         :                                                 :         :........25  DEPTH OF HORIZ PROJECTING ELEMENT
   :         :                                                 :         :........26  FIRE RES OF OPGS IN EXT WALL AT A S WALL
   :         :                                                 :........27  AREA SEP WALL EXTENDS TO FOUNDATION
   :         :                                                 :........28  ACTUAL TOP OF AREA SEP WALL
   :         :                                                 :........29  REQUIRED TOP OF AREA SEP WALL
   :         :                                                 :         :.......-17  FIRE RESISTANCE OF AREA SEP WALL
   :         :                                                 :         :........30  FIRE RESISTANCE OF ENTIRE ROOF
   :         :                                                 :         :........31  FIRE RES OF ROOF WITHIN 5 FT OF A S WALL
   :         :                                                 :         :........32  COMBUSTIBILITY OF ROOF CONSTRUCTION
   :         :                                                 :         :........33  AREA SEP WALL SEP BLDGS WITH DIFF HEIGHT
   :         :                                                 :         :........34  FIRE RES OF EXT WALL 10 FT FROM LOW ROOF
   :         :                                                 :         :........35  FIRE RES OF OPGS IN EXT WALL AT LOW ROOF
   :         :                                                 :         :........36  FIR RES OF LOW ROOF 10 FT FROM A S WALL
   :         :                                                 :         :........37  OPGS PRESENT IN LOW ROOF 10 FT FROM A S WALL
   :         :                                                 :........38  AIR DUCTS PIERCE AREA SEP WALLS
   :         :                                                 :........39  FIRE DAMPERS IN AIR DUCTS MEET UBC STD 43-7
   :         :.........6  TOTAL ALLOWABLE STORY AREA
   :                   :.........7  ALLOWABLE AREA FROM TABLE 5C
   :                   :         :...............................................-16  CONSTRUCTION TYPE
   :                   :         :...............................................-44  OCCUPANCY GROUP
   :                   :.........8  ALLOWABLE AREA MULTIPLIER FROM SECTION 506
   :                   :         :........40  ALLOWABLE AREA INCREASE FOR SEPARATION
   :                   :         :         :........41  NUMBER OF SIDES WITH SEPARATION MORE THAN 20 FT
   :                   :         :         :........42  MINIMUM WIDTH OF SEPARATION
   :                   :         :         :........43  NUMBER OF STORIES
   :                   :         :         :.....................................-44  OCCUPANCY GROUP
   :                   :         :         :..................45  OCCUPANCY NAME
   :                   :         :........46  ALLOWABLE AREA INCREASE FOR OCCUPANCY
   :                   :         :         :.....................................-16  CONSTRUCTION TYPE
   :                   :         :         :.......-41  NUMBER OF SIDES WITH SEPARATION MORE THAN 20 FT
   :                   :         :         :.......-42  MINIMUM WIDTH OF SEPARATION
   :                   :         :         :.......-43  NUMBER OF STORIES
   :                   :         :         :.....................................-44  OCCUPANCY GROUP
   :                   :         :         :........48  FIRE EXTINGUISHING SYSTEM PRESENT
   :                   :         :........47  ALLOWABLE AREA INCREASE FOR FIRE EXT SYS
   :                   :                   :.........5  ACTUAL AREA OF TOTAL BUILDING
```

Figure 2.6 Computer Printed Information Network

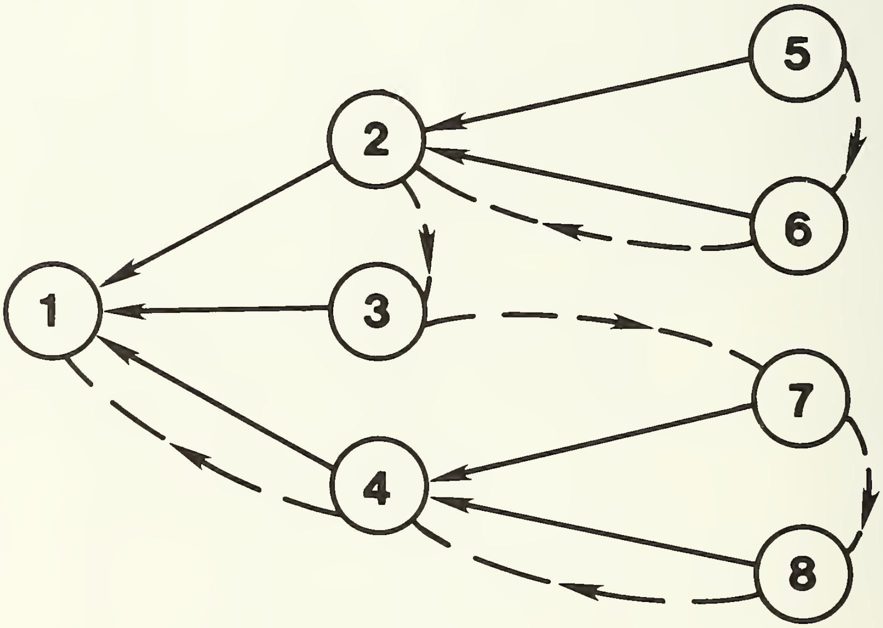
```
:                   :                   :         :......-14* AREA SEPARATION WALL CHECK
:                   :                   :         :..........................-16  CONSTRUCTION TYPE
:                   :                   :         :..........................-44  OCCUPANCY GROUP
:                   :                   :         :......-45  OCCUPANCY NAME
:                   :                   :         :.......51  HEIGHT
:                   :                   :         :.......52  FIRE RESISTANCE OF OCCUPANCY SEPARATION
:                   :                   :         :.......72  ALLOWABLE HEIGHT FROM TABLE 5D
:                   :                   :......-43  NUMBER OF STORIES
:                   :                   :....................................-44  OCCUPANCY GROUP
:                   :                   :......-48  FIRE EXTINQUISHING SYSTEM PRESENT
:                   :                   :.......49  FIRE EXT SYS USED TO INCREASE ALLOWABLE HEIGHT
:                   :                   :.......50  FIRE EXT SYS USED TO SUBSTITUTE FOR 1 HOUR FIRE RES
:                   :........9  ALLOWABLE AREA FROM CHAPTER 16
:                   :         :.......10  FIRE ZONE
:                   :         :..............................................-16  CONSTRUCTION TYPE
:                   :         :................-43  NUMBER OF STORIES
:                   :         :..............................................-44  OCCUPANCY GROUP
:                   :         :..........................-45  OCCUPANCY NAME
:                   :         :.......73  MAX DISTANCE FROM WALL TO PROPERTY LINE
:                   :         :.......74  EXTERIOR WALL IN ACCORD WITH TABLE 5A
:                   :........................................................-16  CONSTRUCTION TYPE
:                   :..........................-43  NUMBER OF STORIES
:                   :........................................................-44  OCCUPANCY GROUP
:                   :.......54  CONSTRUCTION NAME
:                   :.......55  FIRE RESISTANCE OF ENTIRE BUILDING
:                   :.......56  ACTUAL MAXIMUM EXIT DISTANCE
:                   :.......57  ALLOWABLE EXIT DISTANCE FROM 3302
:                   :.......71  AREA INCREASE FOR FIRE ZONE
:                             :......-10  FIRE ZONE
:........3  TOTAL BUILDING AREA CHECK
:         :.....................................-5* ACTUAL AREA OF TOTAL BUILDING
:         :.......-6* TOTAL ALLOWABLE STORY AREA
:         :.......11  BASEMENT OR CELLAR PRESENT
:         :.......12  ACTUAL AREA OF BASEMENT OR CELLAR
:         :         :....................................-14* AREA SEPARATION WALL CHECK
:         :.......13  BASEMENT OR CELLAR QUALIFIES AS A STORY
:..............................................-43  NUMBER OF STORIES
:............................................................................-44  OCCUPANCY GROUP
:........................................................-45  OCCUPANCY NAME
:.......53  CHECK FOR AREA OF H OCCUPANCY
:         :.......-4* ACTUAL AREA OF STORY
:         :.....................................-5* ACTUAL AREA OF TOTAL BUILDING
:         :....................................-43  NUMBER OF STORIES
:         :..............................................-45  OCCUPANCY NAME
:         :................-55  FIRE RESISTANCE OF ENTIRE BUILDING
:         :.......58  FIRE RES OF APT HOUSE EXCL NONBRG & NONPTY WALLS
:.......59  AREA CHECK FOR OPEN PARKING GARAGE
          :.......60  AREA CHECK FOR A TIER OF OPEN PARKING GARAGE
                    :.......61  ACTUAL AREA OF A TIER OF OPEN PARKING GARAGE
                    :.......62  ACTUAL AREA OF TOTAL OPEN PARKING GARAGE
                    :.......63  ALLOWABLE AREA OF TIER OF OPEN GARAGE
                    :         :.......64  BASIC AREA FOR TIER OF OPEN GARAGE
                    :         :         :....................................-16  CONSTRUCTION TYPE
                    :         :.......67  PERCENT OF PERIMETER OF OPEN PKG GAR WITH OPEN SIDES
                    :.......65  ACTUAL NUMBER OF TIERS IN OPEN PARKING GARAGE
                    :.......66  ALLOWABLE NUMBER OF TIERS IN OPEN PARKING GARAGE
                    :.......68  NUMBER OF SIDES OF OPEN PKG GAR WITH QUALIFIED OPENINGS
                    :.......69  MAX HORIZ DISTANCE FROM NEAREST QUAL OPENING
                    :.......70  STANDPIPE PROVIDED ON TIER
```

Figure 2.6 (continued)



Sequence: 1,2,5,6,3,4,7,8

Figure 2.7 Conditional Ordering of the Nodes on a Network (Knuth's Preorder)



Sequence: 5,6,2,3,7,8,4,1

Figure 2.8 Direct Ordering of the Nodes on a Network (Knuth's Postorder)

Both orders are useful in conventional expression of provisions. Research on verbal discourse [26] indicates a preference for following network traversal of some type when explaining concepts, because it minimizes the burden on the listener. Direct ordering can be compared to a recipe in which all the ingredients are listed and then the various steps to be accomplished are described. In a standard, direct ordering occurs when all terms are defined immediately before they are used. Such ordering is sometimes efficient and appropriate for a short section of detailed and complex provisions, such as a standard test method. However, if it is used for longer sections, it tends to become tedious and boring to read.

Conditional ordering would first define a basic provision, one close to output, and then proceed to define its ingredients. Such ordering is preferable in many situations because it allows the reader who is familiar with the standard to begin reading the most important information and then stop or move on when he reaches the controlling provision, thus increasing his efficiency. It is common for the detailed level of design standards.

Figure 2.9 contains a portion of the information network shown in figure 2.6 and the corresponding outline of provisions written in conditional order. Traversing the spanning trees in either of these orders also allows one to automatically locate all necessary cross references. Each branch that was removed from the original network and replaced with the dash in front of the repeated node simply becomes a cross reference in the narrative.

Wright, Fenves, Nyman, and others [39, 96, 134] described the basic topological operations on the network for both constraint processing and narrative expression. For constraint processing they defined algorithms to seek an ingredient needed for evaluation and to warn all dependents of a change in value, and they described the subscripted data structure necessary for checking a complete project. For the expression of provisions in narrative form they described two styles that follow naturally from the network, conditional and direct, and provided a computer program to display datums in these orders. Most of their algorithms for the traversal of a network were based on the models of Knuth [67], although their definitions of conditional and direct order were not. Nyman and Fenves [97] used a modified network for ordering of text that incorporates some of the logic of the decision tables into the overall network by adding a separate node for each rule. Harris, Wright et. al. [51, 52, 135] have returned to the concept of a one-to-one correspondence between decision tables and nodes on the network, and they have slightly modified the use of the network to produce conditional and direct orders consistent with figure 2.7 and 2.8.

```
Area Check
:
:....Single story area check
:    :
:    :....Actual area of story
:    :    :
:    :    :....Area separation wall check
:    :
:    :....Total allowable story area
:         :
:         :....Table 5C
:         :
:         :....Area multiplier
:         :    :
:         :    :....Increase for separation
:         :    :
:         :    :....Increase for occupancy
:         :    :
:         :    :....Increase for fire extinguishing system
:         :
:         :....Restrictions in Chapter 16
:
:....Total building area check
:
:....Check for area of H occupancy
:
:....Area check for open parking garage
```

(a) abbreviated information network in computer printed format

1 Restrictions on Building Area

2 Allowable Story Area

3 Allowable Area Increases

3.1 Increase for Separation

3.2 Increase for Occupancy

3.3 Increase for Fire Extinguishing System

4 Area Restrictions in Fire Zones 1 and 2

5 Allowable Area for Multistory Buildings

6 Area Restrictions for Multistory Building with H Occupancy

7 Area Restrictions for Open Parking Garages

8 Provisions for Area Separation Walls

(b) outline of provisions in modified conditional order

Figure 2.9 Sample Re-ordering of Provisions

2.4 Organizational System

The decision table deals with the meaning of individual provisions, and the information network captures the precedence relations between provisions. Neither, however, addresses the overall scope and arrangement of a standard. That task is left to a part of the conceptual model called the organizational system.

Methods for organizing a standard in this context are newer and less well developed than the parts of the model described previously, but models did exist prior to this study that do provide tools for examining alternative arrangements of a set of provisions [51, 96, 97]. Because the principal work in this study is directed at improving the organization of standards, these models serve as the starting point.

The organizational systems of past studies have dealt with only the ordering of provisions. Other aspects of the organization described in section 1.2 have not been addressed.

The overall network of decision tables that originally contained switching, testing, and working tables was actually divided into two parts by Wright, Nyman, and Fenves [136]. The information network has evolved from the portion containing the testing and working tables. The network of switching tables was recast as an organizational network ("outline"), which served the same function of providing access to the applicable provisions, but did not use decision tables. Because each point of branching was traversed simply by the selection of one of a set of mutually exclusive classifiers, the power of the decision table was not necessary.

No general rule is available for the identification of the provisions to be included in the organizational system. Case studies [96, 97] have been carried out in which only the provision represented by output, or terminal, nodes of the information network have been taken. Other case studies [51] have included each derived node in the information network. A minimal rule is that at least all the output nodes must be included, in order to guarantee access to each node in the network.

For those organizational systems that include more than just the output nodes, an important concept applies: the ordering of the outline is independent of any ingredience relations among the provisions [51]. In such situations, the information network is used to provide cross references between related provisions that are separated by the outline.

Once a set of provisions is identified for the organizational system, they are classified. Following this, an outline may be generated, controlled by the hierarchical relations among the classifiers.

Classifiers are words that concisely define the scope of a provision. The classification as a whole should cover the scope of the whole standard. The classification system typically has a hierarchical structure that is shown by grouping the classifiers into trees. Two examples of classifier trees are shown in figure 2.10, one taken from a study of a standard for the design of steel structures and the other from a study of some provisions for stairways taken from a model building code. The specific classifiers associated with a provision are called the arguments of the provision. The classifiers at any one level of a tree that are all connected to a common classifier at a higher level are referred to as siblings. (Thus, in figure 2.10b "Dimensions," "Strength," and "Material" are siblings.)

Two requirements on the classification system exist [39, 51]. Each set of siblings must be:

1) mutually exclusive to guarantee that the selection of a provision will be unique
2) collectively exhaustive to insure complete coverage

Fenves, Rankin, and Tejuja [38] classified two sets of provisions, one performance oriented and the other prescriptive. Although they did not develop outlines from the classifications, they did demonstrate two points very pertinent to the construction of such outlines that were then used in other work [39, 51]. The first point was that a set of classifiers could collectively represent the scope of a standard. The second was that at least two trees of classifiers are required to represent the information contained in provisions: one for the physical entity being addressed and a second for the performance attribute being mandated. They also showed that a tree of common properties was very useful to group the physical entities without requiring great detail in naming.

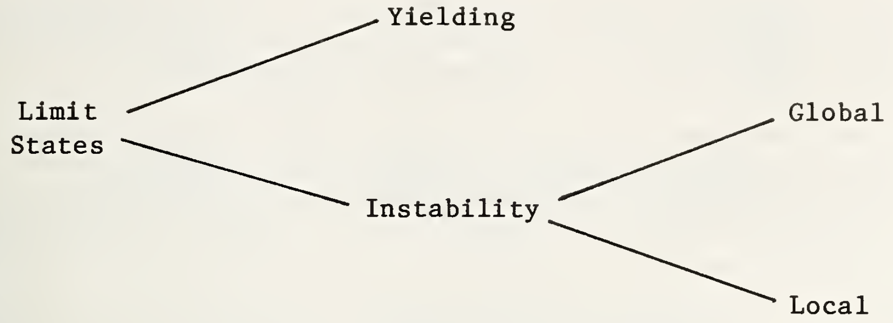
Their work on classification was based on their advancement of the model of a standard into the structure of individual provisions. They showed common underlying structures for two limited categories of provisions, performance requirements and performance criteria. It was these common structures that confirmed the use of two principal trees of classifiers for physical entities and performance attributes. This work also provides a tool for the improvement of the entire model.

The studies that developed outlines have had various requirements on the association of classifiers and provisions. Following is a collection of these requirements:

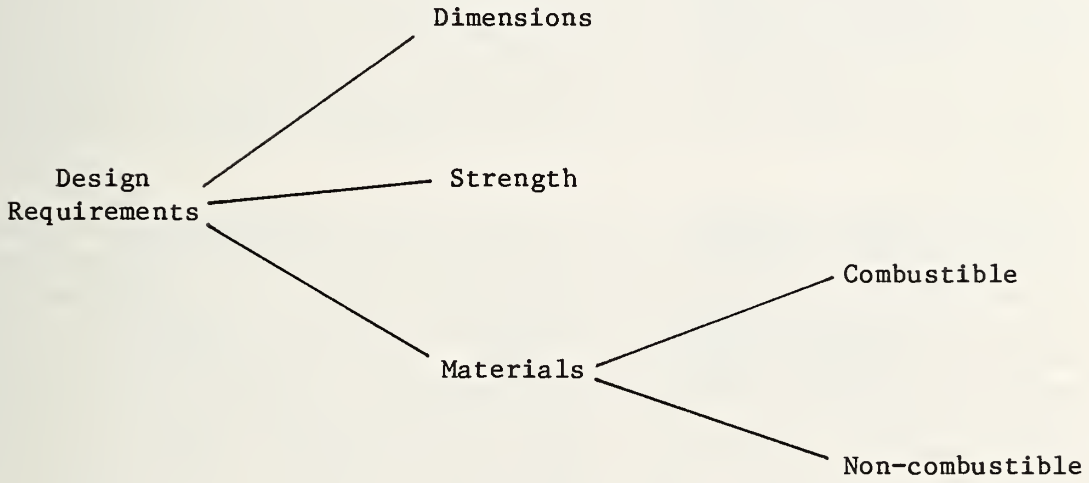
1) Each classifier that is at the end of a branch on its tree must be an argument for at least one provision to insure that the provisions stated cover the scope of the classification.

2) Each provision must have at least one argument to insure that it will be in the outline; some studies [96, 97] have required each provision to have one argument from each tree.

3) In one study [96], more than one provision could have the same set of arguments but this was not permitted in another [97].



a) performance classifiers (behavior) for steel structures, from [96]



b) performance classifiers for stairways, from [51]

Figure 2.10 Examples Trees of Classifiers

4) Most studies allowed one provision to be associated with more than one set of arguments, because it was recognized that some provisions address compound subjects.

5) The classifier from a particular tree that is associated with a provision should be at the lowest level (finest detail) possible--in some studies [38, 51] the more general classifiers that are related to the specific classifier are then implicitly associated with the provision.

The outline, or organization network, is the primary expression of the organization system. It is produced by appending the trees of classifiers onto each other to produce a large tree of classifiers resembling a table of contents. A provision is entered on a branch of the outline when its arguments match the classifiers on the branch. Three algorithms exist [96, 97, 135] for this operation, and they are typically performed on a computer because the operation becomes both complex and tedious for most problems of a practical size. Figure 2.11 and table 2.3 show a very simple example constructed for illustrative purposes only.

The existing algorithms are examined in more detail in subsequent chapters as improvements to the organizational system are discussed. The variations in table 2.3 indicate a few of the features of interest. Outlines b, c, and d show three ways of entering the "Local Buckling" provision: in b the provision is entered quite naturally, but this is not possible when the trees are appended in a different order, so special approaches must be taken. In c the classifier "Instability" is added to the outline at the same level as "Beam" and "Column" (the approach of reference 97), while in d the provision "Local Buckling" is entered on branches with classifiers ("Beam", "Column") that are not among its arguments. The virtues and disadvantages of the approaches are of immediate interest in this study.

In a typical standard, provisions do not exist for all combination of classifiers, in part because only some combinations represent real situations. Thus a complete expression of the trees in a given permutation, such as was done in the simple example of table 2.3, will produce many headings for which no provision exists. "Classifier driven" techniques generate headings representing all the possible combinations of classifiers, although they may print only those for which a provision exists. Such a method is conceptually simple, but requires a great deal of storage space (in a computer). "Provision driven" techniques append classifier trees only when a provision exists with classifiers from both trees. This method reduces the storage space required, but increases the complexity of the decision making process in the generation of the outline. All existing algorithms for outlining are of the second type, but some computer programs [96, 97] actually store the classification system in a fashion that includes all the headings generated in the first type.

An additional level of control that can be imposed on outline generation is the use of priority among the classifiers associated with a provision

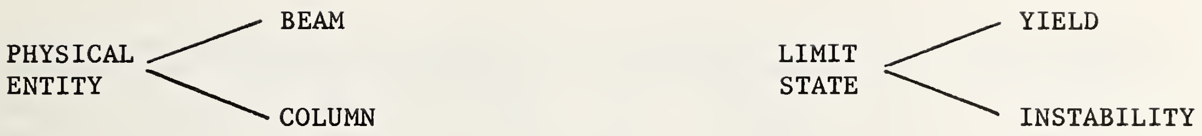


Figure 2.11 Classifier Trees for Table 2.3

Table 2.3 Simple Outline Example

a) provision names and arguments

PLASTIC MOMENT CAPACITY	-	BEAM, YIELD
LATERAL TORSIONAL BUCKLING	-	BEAM, INSTABILITY
AXIAL FORCE CAPACITY	-	COLUMN, YIELD
EULER BUCKLING	-	COLUMN, INSTABILITY
LOCAL BUCKLING	-	PHYSICAL ENTITY, INSTABILITY

b) first possible outline

YIELD		
	BEAM	PLASTIC MOMENT CAPACITY
	COLUMN	AXIAL FORCE CAPACITY
INSTABILITY		LOCAL BUCKLING
	BEAM	LATERIAL TORSIONAL BUCKLING
	COLUMN	EULER BUCKLING

c) second possible outline

BEAM		
	YIELD	PLASTIC MOMENT CAPACITY
	INSTABILITY	LATERAL TORSIONAL BUCKLING
COLUMN		
	YIELD	AXIAL FORCE CAPACITY
	INSTABILITY	EULER BUCKLING
INSTABILITY		LOCAL BUCKLING

d) third possible outline

BEAM		
	YIELD	PLASTIC MOMENT CAPCITY
	INSTABILITY	LATERAL TORSIONAL BUCKLING LOCAL BUCKLING
COLUMN		
	YIELD	AXIAL FORCE CAPACITY
	INSTABILITY	EULER BUCKLING LOCAL BUCKLING

sion [135]. This allows an outline of classifiers to be generated that can possibly contain all permutations of appending trees (for example table 2.3 b and d joined together). Even though more than one branch can have the same set of classifiers, it is possible to retain a unique position for each provision because of the ranking of classifiers associated with it. See table 2.4 for a hypothetical example, and note, for example, that the provision "Euler Buckling" is entered at "Column Instability," not "Instability Column."

If more than one tree of classifiers is available, more than one outline can be generated and compared for quality, because multiple trees of classifiers can be combined to produce an outline in more than one way. Several case studies have been carried out using three trees, which greatly increases the number of alternative outlines that can be generated. Past studies [97] have cited minimum redundancy and pertinence to the intended reader as measures of quality to use in discriminating between alternative organizational networks.

Table 2.4 Hypothetical Example of Outlining with Priority Classification

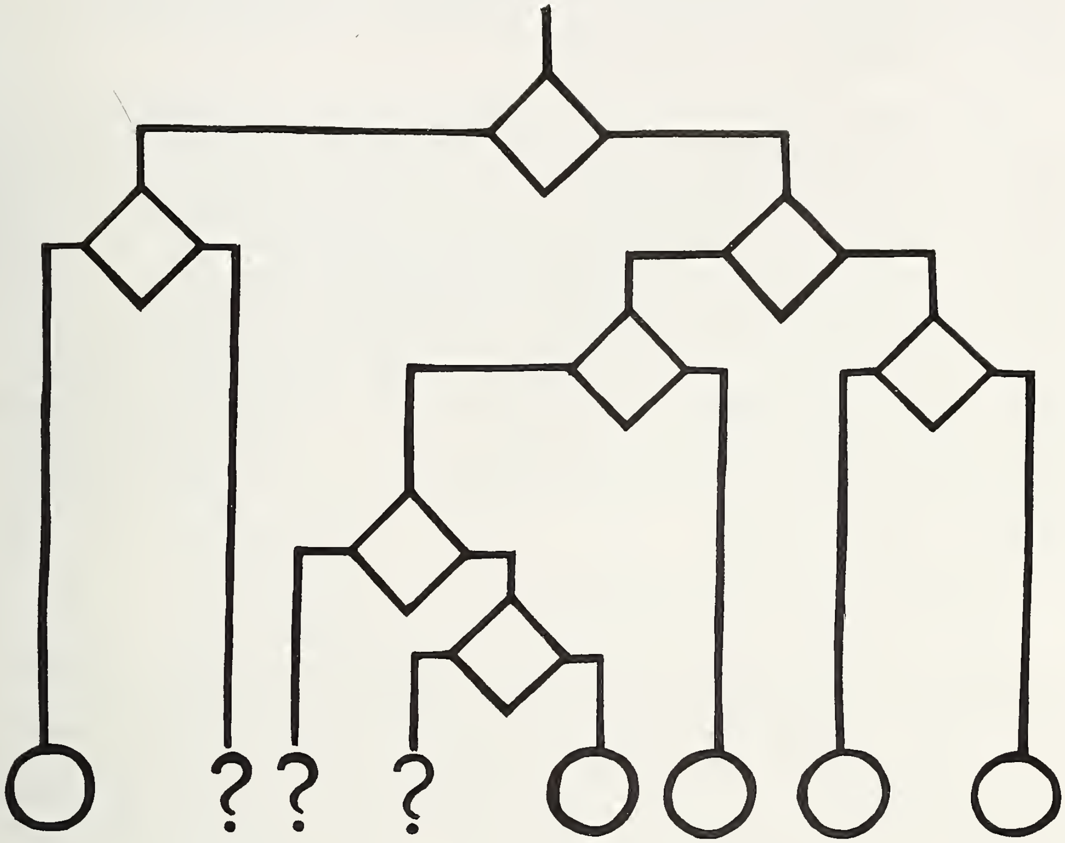
a) provision names and arguments in rank order

		(1)	(2)
PLASTIC MOMENT CAPACITY	-	YIELD	BEAM
LATERAL TORSIONAL BUCKLING	-	BEAM	INSTABILITY
AXIAL FORCE CAPACITY	-	YIELD	COLUMN
EULER BUCKLING	-	COLUMN	INSTABILITY
LOCAL BUCKLING	-	INSTABILITY	---

b) outline for provisions

```
BEAM
   YIELD
   INSTABILITY . . . . . . . LATERAL TORSIONAL BUCKLING
COLUMN
   YIELD
   INSTABILITY . . . . . . . EULER BUCKLING
YIELD
   BEAM  . . . . . . . . . . PLASTIC MOMENT CAPACITY
   COLUMN  . . . . . . . . . AXIAL FORCE CAPACITY
INSTABILITY  . . . . . . . . LOCAL BUCKLING
   BEAM
   COLUMN
```

Facing page: The decision tree provides a convenient logical analysis for a provision.



CHAPTER 3

OPPORTUNITIES FOR IMPROVEMENT OF THE EXISTING MODELS FOR ORGANIZATION

The existing models for expressing the overall organization of a standard lack the rigor of the models for expressing the meaning of provisions and the precedence between provisions. It would be unrealistic to expect that a model for outlining would be as precise as a decision table, because a decision table represents meaning in a form independent of narrative expression or of the order of conditions, actions, or rules, whereas an outline is merely one of many possible arrangements for a given set of information and thus is more subject to individual likes and dislikes. However, a model for organization should provide the qualities presented as objectives in section 1.2 (Relevant, Meaningful, Unique, Complete, Graded, Progressive, Intelligible, Minimal, and Even) and give reproducible results. Failure to meet these objectives will seriously hinder the widespread application and use of the overall conceptual model of standards presented in chapter 2. Therefore, this study of the organization of standards is undertaken.

This chapter deals with four important steps in the study: an analysis of shortcomings perceived in the existing model for organization, particularly in reference to the qualities found in a good organization as presented in section 1.2, brief discussions of two pertinent fields of science, classification and linguistics, that offer important bases for improvements in the model, and a summary of the issues that underlie the problems and potential improvements. Chapters 4, 5, and 6 then present the fundamentals of an improved model for organization and examples of their application. Not all of the problems identified in this chapter are satisfactorily addressed by the improved model; additional research is necessary. And no claim is made that this chapter identifies all the problems with the existing or new models -- further experience with the application of the methods may identify other problems.

3.1 Problems with Existing Models

There are six existing case studies in the classification of provisions in the context of the model described in section 2.4. All of these case studies were classifications of an existing set of provisions; none was the purely "top down" activity of developing a new set of provisions. In four of these studies new outlines were created for the provisions. All of these case studies more or less followed the basic rules described in section 2.4. Classifiers from these studies are shown in tables 3.1 through 3.6 in outline fashion. The observations that follow focus on perceived difficulties in order to identify rational improvements. There is no intent to be critical of any study or investigator. Indeed, Alexander's thesis on the design of an environment -- good fit of a form to its context is not easily recognized, but misfit ordinarily is recognized [4] -- is applicable to the model of the organization of standards.

3.1.1 Scope of the Organizational Model

The existing case studies are not consistent in the definition of what is to be organized. One of the studies [96] included only the datums representing the output nodes of the information network (the "terminal criteria"). Others [97, 98] have included all the "subcriteria," defined as the conditions of the decision tables for the output nodes, thus representing those datums at one level from output in the information network. One [51] has included all of the derived datums, that is all nodes except input nodes. The one study dealing with a performance standard [38] included all the performance requirements and performance criteria but excluded the provisions for evaluation procedures.

Table 3.1 Classification for the AISC Design Specification (Steel) Taken from Reference 96

- Components
  - Members
    - Thin-wall Shape
    - Solid Shapes
    - Encased Components
    - Non-encased Components
  - Elements of Members
    - Bearing Stiffeners
    - Beam/Girder Web
  - Connections
  - Connector
    - Bolts
    - Rivets
    - Groove Weld
    - Plug/Slot Weld
    - Fillet Weld
    - Shear Connector
    - Bearing Surface
    - Pins
    - Roller/Rocker

- Limit States
  - Yielding
  - Instability
    - Global
    - Local

- Stress States
  - Axial Force
    - Tension
    - Compression
  - Flexural Force
  - Shearing Force
  - Torsional Force
  - Bearing
  - Combined Forces
    - Comprression + Bending
    - Tension + Bending
    - Shear + Bending

Table 3.2 Classification for the AISC Design Specification (Steel) Taken from Reference 97

- Components
  - Steel Member
    - Member Other than Beam or Girder
    - Beam Segment in Region of Last Hinge
    - Beam Segment Not Adjacent to Plastic Hinge
  - Element of a Member
    - Stiffened of Unstiffened Elements
    - Web
      - Part of Hybrid Girder
    - Flange
    - Stiffeners
  - Composite Member
    - Non-Encased
      - Shoring Provided
      - No Shoring Provided
    - Encased
  - Vertical Bracing
    - Beams or Girders Used as Bracing
    - Typical Diagonal Bracing

- Limit States
  - Yield
  - Ultimate Capacity
  - Instability
    - Local Buckling
    - Overall Member Buckling
    - Lateral-Torsional Buckling
  - Exessive Slenderness Ratio
    - X-Axis
    - Y-Axis
  - Horizontal Shear Capacity

- Stress States
  - Axial Force
    - Tension
    - Compression
  - Shear
    - Horizontal
  - Bending
    - Compressive Stress Due to Bending
    - Tensile Stress Due to Bending
  - Combined Stress
    - Compression + Bending
    - Tension + Bending
    - Tension Field Action (Shear + Tension)

Table 3.3 Classification for the LRFD Design Criteria (Steel) Taken from Reference 97

- Components
  - Member
    - Beam
      - Wide-Flange or Channel
      - Doubly Symmetric Wide-Flange
      - Solid Section (Not Rectangular)
      - Rectangular Solid Section
      - Box-Shape Section
      - Hybrid Wide-Flange Section
    - Plate-Girder
      - Web
      - Stiffener
    - Composite Section

- Limit States
  - Yield
  - Excessive Slenderness
  - Overall Column Buckling
  - Plate Local Buckling
  - Lateral-Torsional Buckling
  - Flange Local Buckling
  - Web Local Buckling
  - Plastic Moment Capacity
    - About Major Axis
    - About Minor Axis
  - Web Buckling
  - Tension Flange Yield
  - Geometrical Requirements
    - Excessive Web Slenderness
    - Stiffener Spacing
    - Stiffener Size
  - Ultimate Capacity

- Stress States
  - Axial Force
    - Tension
    - Compression
  - Shear
  - Bending
    - About Major Axis
    - About Minor Axis
  - Combined Stress
    - Flexure + Shear
    - Flexure + Tension
      - Uniaxial Bending
      - Biaxial Bending
    - Flexure + Compression
      - Uniaxial Bending
      - Biaxial Bending
    - Combined Stress: Flexure, Shear, Torsion, & Axial Force

Table 3.4 Classification for the BOCA Stair Provisions Taken from Reference 51

Types of Interior Stairways			(Function)
	Required Exit		
	Supplementary Exit		
	Other		
Design Requirements			(Performance)
	Dimensions		
	Strength		
	Materials		
		Combustible	
		Noncombustible	
Appurtenances			(Physical)
	Landing Platforms		
	Handrails and Guards		
	Enclosures		
	Doors		

Table 3.5 Classification of the Residential Solar Energy Performance Criteria Taken from Reference 38

Category I: Physical Classification

a) Top Level Table

- System
  - H, HC *
  - HC only *
  - DHW *
- External
  - Drains
  - Supporting Structure
  - Fire Resistant Assemblies
  - Manual

b) Intermediate Level Table

(Subsystems of * items in table a)

- Collector
- Storage
- Transport
- Control
- Auxiliary Energy

c) Detailed Level Table (for each item in table b)

- Mechanical
  - Motive
    - Pump
    - Fan
  - Heat Exchanger
    - Heater
    - Condenser
  - Conveying
    - Piping
      - Manifolds
      - Circulation Loops
      - Catch Basins
      - Drain/Fill Attachments
    - Ducts
  - Hangers
- Electrical
- Structural
  - Conventional
    - Ultimate Strength
    - Working Stress
  - Non-conventional
- Collector
  - Mounting
  - Glazing
  - Cover Plate
- Control
  - Regulators
  - Instrumentation
  - Controllers
  - Shutoff Valves
- Fluids
  - Liquids
    - Hazardous or Toxic
- Filters
- Gaskets and Sealants

Note: "H" means "heating," "HC" meands "combined heating and cooling," and "DHW" means "domestic hot water."

Table 3.5 (Concluded)

Category II: Classification of Attributes

- Functionality
  - Rating
  - Efficiency
- Operability
  - Requisites
  - Dysfunctions
- Habitability
  - Comfort
  - Health
- Maintainability
- Safety
  - Intrinsic
  - Extrinsic

Category III: Classification of Properties

- External Exposure
  - Rain, Hail, and Ice
  - Pollutants, Solar Degradation
  - Freezing
  - Wind
- Internal Exposure or Contact (other than fluids)
  - Elevated Temperature and Pressure
  - Dissimilar Materials
  - Leachates and Decomposition Products
- Exposure to or Contact with Fluids
  - Potable Liquids
  - Non-Potable Liquids
  - Corrosive Liquids
  - Pressurized Liquids
  - Liquids Requiring Special Handling
  - Air
- Location
  - Elevated
  - Buried
- Access
  - Requiring Access
  - Accessible to Occupant
- Mechanical
  - Subject to Vibration
  - Involving Moving Parts
- Structural
  - Requiring Cutting
- Serving Multiple Housing Units

Table 3.6 Classification of the Plumbing Code Taken from Reference 38

Category I: Physical Classification

a) Systems

Drainage
Vent
Water Distribution

b1) Drainage Subsystems

General
Drainage Piping
Building Sewer
Building Drain
Storm Drain
Storm Sewer
Area Drain
Subsoil Drain
Hoistway Drain
Catch Basin
Gutters
Condensate Drain
Indirect Waste Piping
Pressure Drainage
Utility Drain
Industrial Waste Piping
Commercial Waste Piping

b2) Vent Subsystems

General
Vent Piping
Stack Vent
Common Vent
Wet Vent
Relief Vent
Combination Waste and Vent
Circuit and Loop Vent

c1) Drainage Components

General
Interceptors
Grease Interceptors
Garage Interceptors
Building Traps
Sand Interceptors
Traps
Direct Waste Connections
Indirect Waste Connections
Interior Leaders or Downspouts
Cleanouts
Backwater Valves
Indirect Connections

c2) Vent Components

General
Vent Header
Vent Terminal
Vent Extension
Vent Fittings
Vent Opening
Vent Connection
Waste Pipe
Trap
Cleanouts

Table 3.6 - Category I (continued)

b3) Water Distribution Subsystems

General
Water Piping and Fitting
Standpipe
Hot Water Piping
Water Storage Tank
Hot Water Storage Tank
Hot Water Heater
Water Pump

d) Assembly

General
Bathtub
Lavatory
Shower
Kitchen Sink
  Garbage Disposal
Water Closet
  Flushometer Valve
  Flush Tank
Urinal
  Flushometer Valve
  Flush Tank
Floor Drain
Laundry Tray
Dishwasher
Clothes Washer
Drinking Fountain
Swimming Pool
Hot Water Heater
Sumps
Ejectors, Pumps
Lawn Sprinklers
Interceptors, Intercepting Tanks
Evaporative Cooler, Air Washer, Refrigerator and Cooling Counters
Walk-in Refrigerator
Receptors
Stills, Sterilizers
Condensor
Plumbing Fixtures (unclassified)
Commercial Sinks
Island Sinks

c3) Water Distribution Components

General
Ballcock
Direct Connections
Indirect Connections
Water Storage Control Valves
Gate Valve
Meters
Vacuum Breakers
Pressure Relief Valves
Temperature Relief Valve
Emergency Shut-off Devices
Valves
Covers
Pressure Regulators
Pressure--Temperature Relief Valve

e) Element

General
Piping (discharge)
Piping (supply)
Drainage Fittings
Shower Receptors
Faucet
Shower Head
Drains
Traps
Vents
Valves
Mixing Valves
Gate Valve
Vacuum Breakers
Washdown Pipe
Cleanouts
Hose Bib Outlet
Outlets
Inlets
Strainers
Flashings
Backflow Preventing Devices
Motors, Compressors, and Air Tanks
Trap Arm
Overflows
Ballcock
Bends

Table 3.6 - Category I - (Concluded)

f) Appurtenances

General
Threads
Connections (direct)
Joints
Flanges
Plugs
Locks
Bushings
Hangers
Supports
Slip Joints
Expansion Joints
Hose Clamps
Unions
Ferrules
Screws
Bolts
Washers
Access Covers
Outlets
Joining Material
Fittings
Connections (indirect)
Cleanouts
Seals
Flashings

g) Materials

General
Wrought Iron
Steel
Brass
Copper
Galvanized Iron
Cast Iron
Bronze
Lead
PVC (Plastic)
ABC (Plastic)
Malleable Iron
Concrete
Metal

Category II: Attributes

Function
Health and Safety
Operability, Durability, Reliability
Maintainability
Standards

In fact the inconsistency may be deeper than it first appears, because the definition of an output node, or terminal criterion, is not completely consistent, either. The study of performance standards [38] stated that the element of a prescriptive or procedural standard that would be equivalent to a performance requirement or criterion was a paragraph, making no reference to the position in an information network. In some of the case studies, the existing organization (that is, the "paragraphs") strongly influenced the definition of terminal criteria, while in others cross references between paragraphs were included in the decision tables and information network, thus tending to reduce the number of terminal criteria. These latter studies are the same ones that tended to go "further" into the information network with the organizational studies.

A consistent basis for deciding what to classify is needed, and this basis must include an adequate consideration of the interface of the organizational model with datum definition and the information network.

3.1.2 Basic Classes for a Classification

It is instructive to note that there are significant similarities in the basic classes used in the six studies. Each of the six has one class that names the objects covered by the standard. This class is given the name "component," "appurtenance" or "physical." However, there are significant differences in the basic classes also. Three of the examples have a class for "limit states," which is defined as a state of incipient unsatisfactory behavior in a specific mode [133]. Two of the examples have a class for "attributes," which are, by examination of the tables, qualities necessary for the proper performance of the physical objects that are covered by the provisions. Limit states and attributes are related, but by no means are the same. Likewise the class "stress state" is similar to the class "function" and the class "property," but there are significant differences, which may not be obvious. Table 3.7 summarizes these similar classes.

With the exception of two case studies (tables 3.5 and 3.6), the existing studies are nearly silent on the selection of the basic classes, thus giving little guidance for future applications. In those two Fenves, Rankin, and Tejuja [38] make a strong point for classifying provisions according to the physical entity addressed and the performance attribute desired. (A physical entity is an object or a system of objects and a performance attribute is a quality that the physical entity must provide for its human users.) The establishment of a set of basic classes is important because it gives a starting point for the development of a relevant and meaningful classification system for a particular standard.

Table 3.7 Summary of Basic Categories from Referenced Case Studies

	Generally Similar Categories		
Source Table and Short Name	Classes of Objects	Classes Related to Performance	Classes Related to Function
3.1 (AISC#1)	Components	Limit States	Stress State
3.2 (AISC#2)	Components	Limit States	Stress State
3.3 (LRFD)	Components	Limit States	Stress State
3.4 (Stairs)	Appurtenances	Design Requirements	Type of Stair
3.5 (Solar)	Physical	Attributes	*
3.6 (Plumbing)	Physical	Attributes	---

* The class "property" has something in common with the other classes in this column, but it also includes many classifiers not related to function.

3.1.3 Rules for the Basic Structure of a Classification

The existing case studies reveal some problems with the application of the basic rules described in section 2.4. For example, each of them violate in some fashion the rules for mutually exclusive (unique) and collectively exhaustive (complete) sets of classifiers. Table 3.2 shows "steel members" and "vertical bracing" as siblings, yet they are not mutually exclusive. The same problem exists for "cover plate" and "glazing" in table 3.5 and for "water piping and fitting" and "hot water piping" in table 3.6. Although one must study the provisions to do so, it can be shown that one of the "design requirements," ("slip resistance") is missing from table 3.4. Likewise "column" appears to be a missing component in table 3.3 because the limit state "overall column buckling" is present. The failure to satisfy these two rules raises questions about the need for the rules, the consequence of failing to meet them, and the feasibility of constructing classifications in accordance with them.

Furthermore, the existing case studies raise questions as to whether additional rules are needed in the development of classifications. Some are developed strictly as trees, for example tables 3.1 through 3.4, while others are developed in a looser, but still hierarchical, arrangement, such as the "Physical" classification in tables 3.5 and 3.6. Obviously the latter arrangement develops many more possible combinations of clasifiers, but it would be possible to do the same with a larger set of smaller tree-like classes, one for each level of the hierarchical class. (The difference between a tree and a "loose" (non-nested) hierarchy is discussed in section 3.2.1.) The use of a loose hierarchy may be a way of showing that there are precedence relations among the classes. It is difficult to proceed in the actual classification and outlining of provisions without answers to such questions.

3.1.4 Association of Classifiers and Provisions

The problem of relevant and meaningful associations between provisions and their arguments is not entirely separable from the problem of formulating a classification. However the concept of association of arguments to provisions does provide a convenient framework for discussing some additional problems. In some studies [97], the rules for association of arguments to provisions attempt to provide for the unique classification of each provision, thus promoting unambiguous access. However, these rules are not universally followed in the studies, and the need for them is not entirely clear.

No rules exist to assure that classifiers (and thus headings) that are associated with a provision are relevant and meaningful. Meaningful and, to a great extent, relevant are qualities of association that are a matter of degree rather than kind. Nevertheless, an improvement over the present subjective basis for association is called for. Table 3.8 shows several examples of the classification

Table 3.8 Examples of Perceived Incorrect Classing of Provisions

Table of Classifiers	Provision	Classifiers	Remarks
3.1 [96]	"Rivets and bolts subject to combined shear and tension shall be so proportioned, that . . ." [119]	Rivets, Shear; Bolts, Shear; Rivets, Tension; & Bolts, Tension. (It is classed four separate ways.)	A class exists for combined bending and tension, but no class exists for combined shear and tension, which would be appropriate for this provision.
3.2 [97]	(It is not possible to quote the provision directly. The provision is actually a condition in a decision table that tests a limit on the allowable tensile bending stress. The limit depends on several formulas, none of which are related to member buckling.)	Member, Ultimate Capacity, Bending-Tension.	"Ultimate Capacity" is meant to include both yielding and buckling, but since this criterion applies only to tension members, "Yielding" would appear more appropriate.
3.4 [51]	"Winders shall not be permitted in required exitway stairways except that in one- and two-family dwellings and in ornamental stairways not required as an element of an exitway, treads with a minimum . . ."	Other	"Required Exit" appears to be a more relevant functional classifier than "Other".
3.5 [38]	"The structural elements and connections of the H, HC, and DHW systems shall not fail under the ultimate loads expected during the service life of the structure." [59]	System-Structural, Operability-Dysfunction.	"Safety-Intrinsic" appears to be at least as relevant in attribute as "Operability-Dysfunction."
3.6 [38]	"Joints at the roof around vent pipes, shall be made watertight by the use of approved flashings or flashing material." [127]	Venting, Vent Piping, Maintainability; and Venting, Standards. (It is classed two separate ways.)	"Operability, Durability, and Reliability" seems as appropriate for an attribute as either "Maintainability" or Standards."

of provisions taken from the studies referred to in section 3.1. Simply pointing out that classing the provisions differently appears to increase both the relevance and the meaning is enough to demonstrate the lack of rigor available. The goal of reproducibility in classing has not been reached. These examples also show the problems that a reader of a standard has in locating the appropriate provisions. The work of Fenves, Rankin and Tejuja [38] indicates a promising way of establishing the relevance of classifiers.

3.1.5 Assurance of Completeness

All of the case studies referred to in previous sections have dealt with classing an existing set of provisions; four of these have also dealt with the rearrangement of the provisions. One of the objectives set forth in section 1.2 is Completeness. There are good reasons for wanting a set of classifiers to be complete.

Although none of the existing case studies did so, an index can be developed directly from the classifiers and provisions. An index is a valuable tool for access. For reliable access from an index, a provision should be classed in all relevant ways. Since none of the case studies developed an index, it is not possible to infer much from them as far as the desired completeness of a classification for indexing. It is likely, however, that the method used in the study reflected in table 3.4 would be found wanting, because most provisions in that study were only classed according to one of the basic trees. In all the other case studies each provision was classed according to at least two basic trees, and in several studies each provision was classed according to each tree of the classification, giving at least two entries for the provision in an index.

The model for organization also can be applied to the problem of formulating a new standard, although this was not the situation in any of the reported studies. For full benefit of the use of this model in formulation, a different criterion for completeness exists: the classification is used to check that provisions exist for all pertinent combinations of classifiers. (Note that this admits that some combinations of classifiers would not imply pertinent provisions; for example no provision would exist that could be classified as "solid section" and "web local buckling" from table 3.3, because the concept of a web is not meaningful in the normal context of solid sections.)

For the rearrangement of a set of provisions, it is not necessary that a set of classifiers be complete in either of the above senses. It is only necessary that a complete enough classification be used to provide a unique location in a linear outline for each provision. All the existing algorithms for generating outlines have been produced in studies where a tool for the rearrangement of provisions has been a primary goal. Thus it is understandable that these algorithms suppress the generation of headings for which no provision has been assigned a priori. When the emphasis is on generating all the provisions necessary to cover a given scope, then an algorithm for predicting missing provisions is called for. Likewise, when the emphasis is on providing an index for access to provisions, then a motive exists for providing a complete set of classifiers. More guidance is called for in these two situations.

3.1.6 Method for Outlining

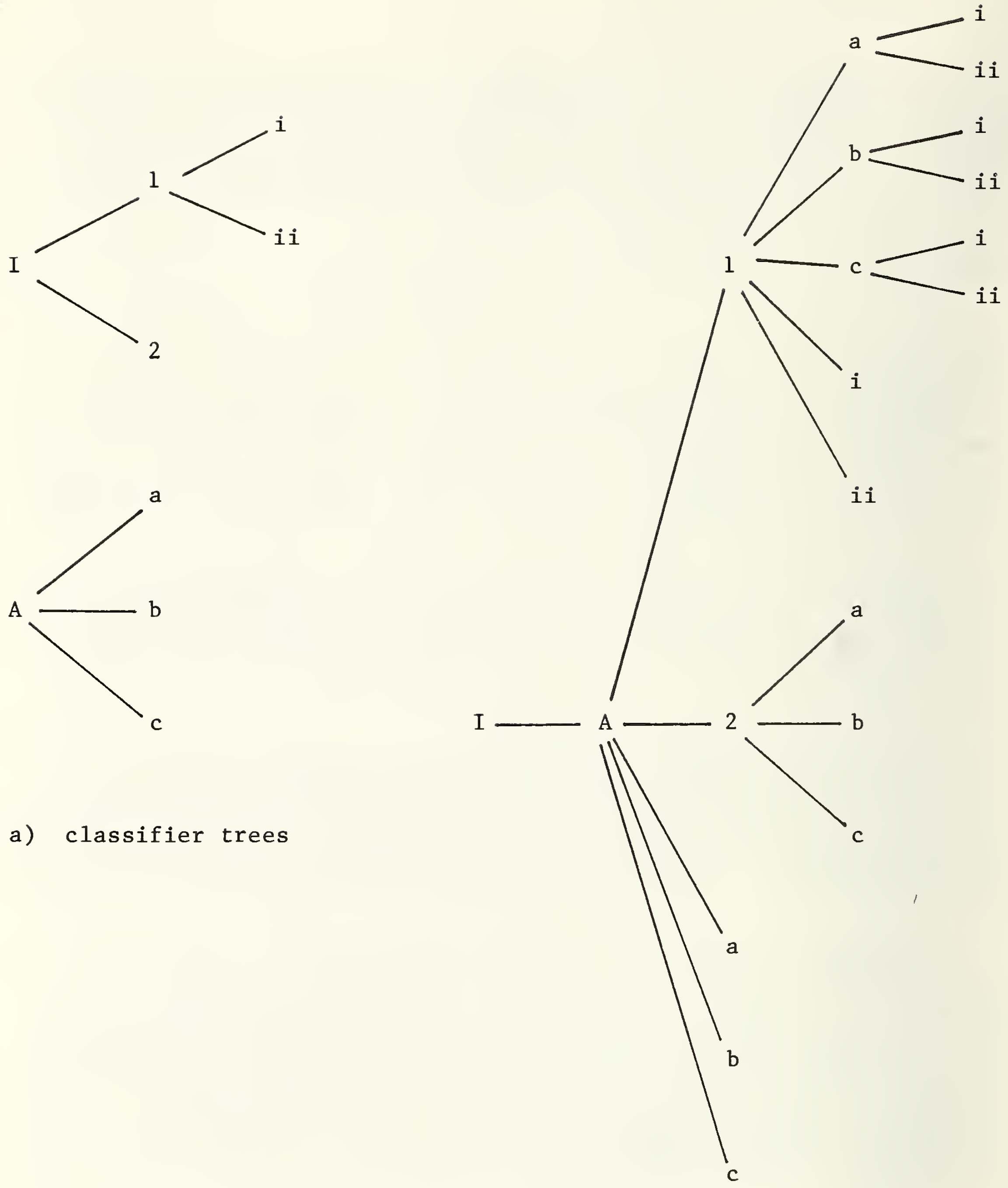
Past studies have used different criteria for tearing and appending trees of classifiers to form outlines. In two algorithms [97, 135], entire trees of classifiers are appended at one time, although one of them [135] liberally deletes classifiers that are not used. In one algorithm [97], the classifer trees are divided into "levels" before the outline is constructed. With this algorithm, it is possible to construct an outline in which classifiers at one level of a tree appear at several different levels in the outline. Figure 3.1 shows a schematic example of the technique. Note that the classifiers "a", "b", and "c" appear on different levels, as do other classifiers. Apparently, this is done to allow proper outlining of provisions that are classified by a "general" classifier from one tree and a "detailed" classifier from another tree (for example "I" and "a").

Guidance is needed for knowing just how far, if at all, a classifier tree may be subdivided before an outline is generated. Guidance is also needed for knowing when classifier trees can and cannot be appended. It would appear that creating an outline with the classifiers "1", "2", "a", "b", and "c" at one level is akin to mixing apples and oranges; the objective Unique may be lost.

The existing algorithms are not identical with respect to the rules for determining when a provision may properly be entered on a branch of an outline. In two of them [96, 97] the arguments of the provision must match each of the classifiers on the branch and no other classifiers may be present, while in the third [135] allowance is made for classifiers that are parents of arguments. Various methods of comparing argument sets to a branch of an outline need to be examined, giving consideration to the objectives stated in section 1.2.

3.1.7 Choice of Alternative Arrangements

Several of the case studies referenced in section 3.1 generated alternative outlines and then proceeded to select a "best" choice. Two qualities were used in a subjective way to make the selection; they relate to the objectives Complete and Meaningful. One study [96] pointed out that the tree of classifiers used to start the outline (the primary tree) should be complete in the sense that all provisions are associated with a classifier from it. If a provision



a) classifier trees

b) fully appended tree in the order 1-a-i

Figure 3.1 Schematic Example of Tearing and Appending Trees of Classifiers Following Reference 97

were not associated with the primary tree it might not have any relevant location in the outline, or it might have a multitude of possible locations within the outline. Tables 2.3 c and d show both types of problems. Nyman and Fenves [97] discriminate between two outlines by choosing the one with the classifiers most meaningful to the users at the highest level in the outline (that is, the major subdivision).

One criterion for selection used explicitly in previous work is that the number of headings should be a minimum [96]. This agrees well with one of the desirable qualities for organization stated in section 1.2. Table 3.9 shows a hypothetical example of two outlines of headings without provisions produced by reversing the order of the two trees. All other things being equal, the outline with the smaller number of headings would be more desirable because it is more compact. This, however, is a weak criterion for the quality of an organization, even for choosing between organizations that achieve the more basic objectives of Completeness and Unique. It does not measure other pertinent qualities that make possible the quick and correct access to provisions.

There is a need for objective measures for Relevant, Unique, and Complete and for other qualities like Intelligible and Even to compare different outlines of the same provisions.

3.1.8 Aids for Automatic Processing of Provisions

The classification for a standard can and should provide more than an outline and an index. Provisions will not always be accessed in the same order or as a total set. Selective access to provisions is and will be the most common operation. The top level organization must provide access in such situations, for manual use and particularly for use with a computer for automatic processing of constraints.

Early work with decision table representation of standards made use of a network of switching decision tables to provide access in an environment of computer usage [45]. The network of switching tables provided a logical rigor that is not apparent in an outline. By definition and by design, each switching decision table made a choice among a collectively exhaustive set of mutually exclusive "classifiers." The network provided a sequence for accessing provisions if given more than one combination of classifiers. The rigor was "expensive" in that decision tables are more powerful and bulky than necessary to represent such choices, and the network of switching tables did not provide a tool to study the rearrangement of provisions. However, this rigor may be necessary and the switching table concept may be useful. Further study is called for.

It is worthwhile to note that reference 98 provides both a switching network and a classification for the same set of provisions, and they do not correspond. The relation of outlines, organizational

Table 3.9 Hypothetical Example for Illustration of Minimal Permutations

Classifiers

Class 1	Class 2
Dimensions	Landing Platforms
Materials	Handrails and Guards
	Enclosures
	Doors

Outlines

Class 1 as Root

- Dimensions
  - Landing Platforms
  - Handrails and Guards
  - Enclosures
  - Doors
- Materials
  - Landing Platforms
  - Handrails and Guards
  - Enclosures
  - Doors

Class 2 as Root*

- Landing Platforms
  - Dimensions
  - Materials
- Handrails and Guards
  - Dimensions
  - Materials
- Enclosures
  - Dimensions
  - Materials
- Doors
  - Dimensions
  - Materials

*Note that the outline with Class 2 as root contains more headings (is longer) than the other outline.

networks, and classifications to the use of a standard in an automatic processing mode needs to be determined and full advantage should be taken of the work of classification.

3.1.9 Need for More Examples

In addition to the needs identified in the previous sections, there is a need for more examples of classification. There are other resources to consider that can fill these needs. There is considerable information available on the science of classification, as will be discussed in section 3.2. There are also other well established classifications of information that are related to the classification of provisions. Kapsch [64] has summarized 15 systems for classification of architectural information. Several of these are classifications for provisions, and although they are not consistent with the model for organization described in section 2.4, they provide useful references. Widespread use of the method for organization demands a large amount of consistent information, thus any existing classifications that can be used to an advantage should be identified.

3.2 The Science of Classification

Classification plays a major role in the organizational system. A brief discussion of the fundamentals of classification is offered here so that an improved model for organization might take advantage of existing knowledge. Classification is simply the systematic arrangement of things into groups, or classes. The grouping normally is based on properties of the things so that like things are put together and unlike things are kept apart [130]. Classification is basic to human thought--the recognition of similarities in patterns is among the earliest forms of sense perception in living organisms [68]. Its place in science is amply described by Jevons [61], a 19th century British economist and philosopher:

> Science, it has been said at the outset, is the detection of identity, and classification is the placing together... those notions or objects between which identity has been detected. Accordingly, the value of classification is co-extensive with the value of science and general reasoning.

Korner, a modern philosopher, groups distinctions used for mental organization into 3 classes [68, 69]: 1) between an object and its attributes, such as between a column and its height, 2) between an object and its constituent parts, such as between a building frame and the columns within it, and 3) between different classes of objects, such as between columns and beams. The science of classification deals almost entirely with the third class of distinctions, although it must be noted that both the second and third classes depend on the first class for their existence [68]. Biology, library science, and anthropology are examples of branches of science where principles and methods for classification have become relatively highly developed.

3.2.1 Logical Grouping

The basic principles of classification come from set theory. A set of objects is partitioned into two or more subsets (classes) such that two principles are satisfied: 1) no object is in more than one class, and 2) all classes taken together include all objects in the original set. These principles, also called mutual exclusion and collective exhaustion, are known as the logical principles of classification. With varying degrees of rigor, most classifications attempt to meet these two principles. However, as will be pointed out later, these principles do create problems in some instances, and alternative principles sometimes are used.

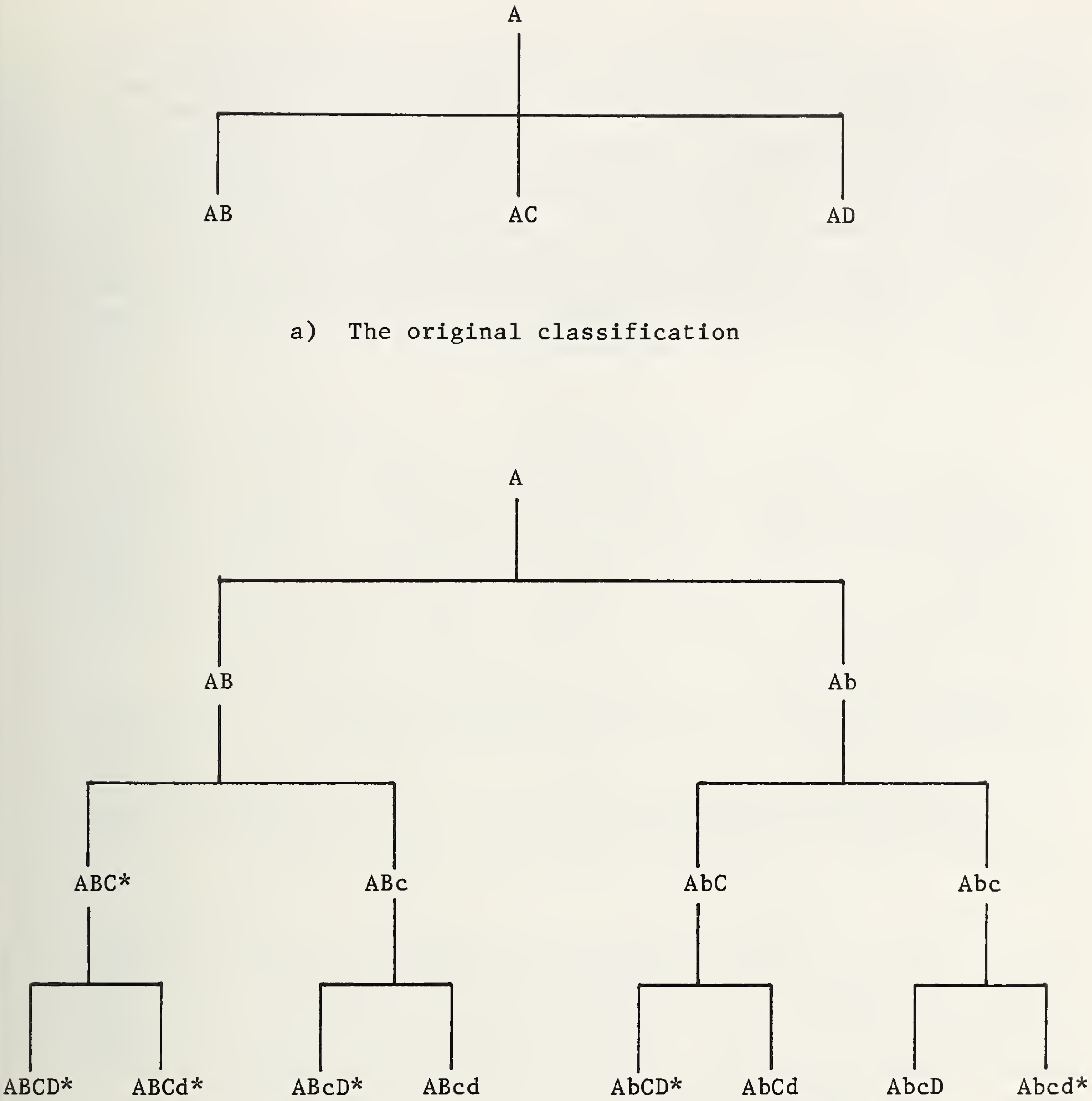
It can be shown that the logical principles of classification may be transformed into logical principles necessary for a characteristic to be used for class selection. To wit: such a characteristic must be possessed by all members of the class and not be possessed by any things not in the class [61].

In general, a classification contains many levels, and the logical principles for dividing a set into a group of classes at one level are extended to cover the subsequent divisions that yield the more detailed classes. The logical principles apply as classes are repeatedly divided to build the classification with many levels. Hierarchical classifications may or may not be strictly logical [117]. Hierarchical means divided into orders or ranks; it means that each level of the classification contains classes with a scope less than the classes in the level immediately above. It does not necessarily mean that any particular class must be a subset of another particular class in the level immediately above. Multi-level classifications that strictly follow the logical principles are characterized by a tree-like, nested structure. Thus, while all of the classifications given in tables 3.1 through 3.6 are hierarchical, the one in table 3.6 is not strictly logical because overlapping can occur. (For example, "trap" occurs in subtables cl, c2, and e.)

A frequently used and sometimes powerful method in classification, which is strongly related to the logical principles, is bifurcate classification, or dichotomy. In this method, each class is divided into exactly two subclasses, the second subclass being defined as all of the parent class that is not contained in the first subclass. Jevons [61] said that it:

> is not only natural and important, but inevitable and the only system which is logically perfect according to the fundamental laws of thought.

Figure 3.2 shows an example from his book in which the division of a class A into three classes, one with the property B (AB), one with property C (AC), and one with property D (AD), is shown to be in error unless it is understood that properties B, C and D cannot co-exist and that each of member of A has one of the three properties.



a) The original classification

b) The complete bifurcate classification, where the upper case letter denotes the presence of the property and the lower case letter denotes the absence of the property. The * indicates classes missing from the original classification.

Figure 3.2 Jevon's Example of Bifurcate Classification

A frequent occurrence in bifurcate classifications is the insignificance of the negative subclasses. This gives rise to a series of of positive subclasses, each with a slightly smaller scope than its parent [13]. An example of such a subordinate series is the first 5 sections of the "Propaedia" of the 15th edition of the Encyclopedia Britannica [30]: 1) Matter and Energy, 2) Earth, 3) Life on Earth, 4) Human Life, and 5) Human Society. A general defect of bifurcate classification is that it becomes cumbersome and unwieldy; in particular, it tends to become very deep. Even Jevons, who professes its logical rigor, states that dichotomy seems to be needless or tedious when the property involved can be measured [61].

3.2.2 Ordering

Once the problem of dividing a set of things into classes has been solved, the remaining problem is the ordering of those classes. Several researchers and developers in the field of classification have considered the problem. True bifurcate classification does not present much of a problem in this aspect, with the usual procedure being to place the positive subclasses first. Furthermore, bifurcation has an influence on the ordering of one other kind of classification. Bliss, a leading American librarian of the early 20th century, identified two kinds of series: subordinate and coordinate [13]. A subordinate series is in reality not a group of siblings produced by logical subdivision, but is the result of a bifurcate classification with insignificant subclasses, such as the example of the Propaedia of the Encyclopedia Britannica presented previously. Although the subordinate series is not a strictly logical classification, it is frequently useful, and it does provide an obvious order.

The most interesting problem is the ordering of a coordinate series. In a coordinate series the individual classes bear a complementary relation to the remainder of the classes, and there is no obvious characteristic remaining for ordering. Vickery, a leading library scientist of today, tabulates [130] the following principles for ordering terms in the class, and thus for ordering subclasses among a class (he attributes the principles to Richardson and Ranganathan [106], also librarians):

1) Logical--from the complex to the simple
2) Geometrical
3) Chronological
4) Genetic--likeness or origin
5) Historical--a combination of the above
6) Evolutionary--from the simple to the complex
7) Dynamic--order of power
8) Alphabetical
9) Mathematical--by means of a notation
10) Decreasing extension
11) Increasing completeness

A simplification of these principles of ordering, which suits the purposes of this study, is that classes in a series may be ordered by a quality different than the quality used for membership in the class or that classes in a series may be ordered by the degree to which they possess the quality from class membership. Hempel, another philosopher, notes that qualities used in the latter sense (to order classes which possess the quality to different degrees) should have several characteristics, which may be condensed to the following: reflexive, antisymmetrical, and transitive [56]. In symbolic terms, these characteristics may be defined for ordering by inequality in the following fashion:

1) reflexive: $x \leq x$
2) antisymmetrical: $x \leq y$ and $y \leq x \Rightarrow x = y$
3) transitive: $x \leq y$ and $y \leq z \Rightarrow x \leq z$

Hempel then goes on to note that there is a significant difference between comparative measurement for partial ordering such as this, and absolute measurement [56].

3.2.3 Philosophy of Classification

Aristotle's "categories" give an early and enduring guide to classification (they can be found in many sources, for example [137]):

1) Substance
2) Quantity
3) Quality
4) Relation
5) Time
6) Place
7) Situation (position)
8) State
9) Action
10) Passivity

It is possible to critize these as failing both the logical principles [86], but they have had an enormous influence on the philosophy of classification, being essentially unchallenged until Kant offered his categories, or "concepts of the understanding" [62]:

1) As to Quantity
   Unity (Measure)
   Plurality (Magnitude)
   Totality (Whole)
2) As to Quality
   Reality
   Negation
   Limitation
3) As to Relation
   Substance
   Cause
   Community
4) As to Modality
   Possibility
   Existence
   Necessity

In the last two centuries, many philosphers have introduced new schemes of ultimate classes; only the novel system of the American C.S. Pierce (the founder of philosophical pragmatism) will be discussed here [44]:

1) Firstness - a thing all by itself
2) Secondness - one thing in relation to another
3) Thirdness - two things "mediated" by a third

He asserted that all higher orders of relations could be reduced to combinations of the three, and that this system was the "most efficient" for organizing concepts in every branch of philosophy [44].

Each of these schemes offers insight into the selection of characteristics with which to define classes. They are not to be thought of as classes themselves. The question of basic classes for building standards is discussed at length in section 4.3.

Strictly logical classifications may be constructed in which the criteria for placing a thing within a certain class are entirely arbitrary [19], however the real worth of a classification generally depends on the significance of the characteristics selected to determine the class membership. Ideally, the characteristics used will have strong correlations with other characteristics so that placement of an object in a class enables one to predict many other characteristics for that object [56, 57]. Mill pointed out that [86]:

> The properties...according to which objects are classified should, if possible, be those which are causes of many other properties.

However, he went on to point out that one usually must select a chief effect rather than the cause as the characteristic for classification. In a classic example of this, Hubble demonstrated that a classification of galaxies based on shape gave much useful information concerning the spectrum of light and radio waves emitted and the constitution of the core, among other things [58].

Many individuals have searched for the one "natural" classification, particularly in biological fields. Such hopes were uplifted by the appearance of Darwin's theory of evolution over a century ago and only comparatively recently have been reassessed, even though Jevons correctly pointed out in 1874 [61] that there is no:

> ...one essentially natural classification which ought to be selected to the exclusion of all other.... There will usually be many other possible arrangements, each valuable in its own way.... There can be no precise distinction between natural and artificial systems.
>
> All arrangements which serve any purpose at all must be more or less natural.

Bliss also pointed out a half century ago in the field of library science that the difference between natural and artificial is only relative, that the real goal is a purposive general system [13]. Hempel goes on to point out that what is natural to one may well be artificial to another, it is a matter of degree [56]. Thus the principles for determining class membership suffer a paradox: there is no one natural, or best, system, yet the closer the system comes to that point, the more significant the characters are, the more generally useful it will be.

Significance (naturalness or relevance) is not the only objective to be met in the selection of characteristics upon which to base a classification. Ranganathan adds two that are quite important: ascertainability and permanence [106]. Ascertainability can also be stated as the following: that the classification must be practical and easy to use (or meaningful, in the terms of section 1.2), for without this, all the foregoing is lost. Metcalf, an Australian librarian, dismisses many elegant subject classifications proposed for libraries thus [84]:

> But we should be able to get along almost without logic and certainly without the very questionable metaphysics of some supposed authorities.... Metaphysical theorizing and bibliosophy have yet to show practical results.

Other writers on the subject of classification set forth similar criteria for judging the quality of a classification: Sneath and Sokal, modern biological taxonomists, call for naturalness, ease of manipulation, and information retrieval as the three goals [117] and Kapsch calls for reproducibility, time independence, and rigor [65].

In summary, it appears that the single most important criterion for a classification is the degree to which it satisfies the purpose for which it is intended. Different purposes give rise to different criteria for judging this overall criterion of goodness. Thus if one of the purposes of the classification would be permanence, then one model which could be examined is Aristotle's classification of the sciences into theoretical (theology, mathematics, and physics), practical (ethics, politics, etc.), and productive (poetics, art, etc.) (reproduced many places, see reference 137 for example), which lasted relatively unchanged for nearly 2,000 years. Changes were not necessary until the Renaissance when new fields of science were developed [130]. If strict rigor is necessary to meet the purpose of a classification, then the characteristic subdivisions may need to follow the rules of bifurcation.

3.2.4 Beyond Logical Classification

Strictly logical classification becomes problematical in at least two situations. The first is the borderline case, something that seems to fit equally well into either of two or more classes. This problem is so pressing in the biological field that a solution has been developed for it. The purpose of biological taxonomy is to develop a large tree-like classification in which each organism may be placed in one and only one class, or taxa.

Biological classification can be divided into three ages [115], the first dating from 1735 when Linnaeus published Systema Naturae. The principal philosophical basis was that the species are unchanging units. Darwin's theory of evolution showed that species are related to each other in a material way through descendants, thus the second age of biological classification. In the current day a third age is emerging that is changing from a tree-like classification of typological units to a population basis in which the classification is no longer a nested tree, however is still hierarchical [115].

This new approach is based on the fact that biological organisms possess a large number of observable, measurable characteristics. The technique of numerical taxonomy uses these characteristics to define classes in a statistical manner [118]. The criteria for class membership is defined by mathematical similarity coefficients that could best be called a measure of family resemblance. Such polythetic classifications are advocated by their sponsors as being more realistic and useful. They account for the borderline case and the extraordinary case (such as classing an animal that has three legs, gives milk, has a leathery hide, and sounds a "moo" as a cow [118]). Numerical taxonomy has been applied in a few fields outside biology in recent years. One application pertinent to this study is that by Kapsch to the classification of old buildings [65], another is the study of communication paths and layouts in offices [20].

The distinction between polythetic classification and most ordinary classifications, which might be called monothetic, is found in the definition of polythetic [117]:

> a group K is defined from a set of G properties, f_1, f_2,...f_n:
> 1) each member possesses a large number of the f_i in G,
> 2) each f_i is possessed by a large number of these members, and
> 3) no f_i is possessed by all of the members of K.

A monothetic classification is one in which all members of a particular class do possess at least one characteristic in common. In polythetic classification, characteristics are usually treated in bifurcate pairs (for example, symmetrical and nonsymmetrical, metallic and nonmetallic, etc.). Two significant limitations on polythetic classification are that each pair of characteristics must apply to all of the members of the population being classified and that the minimum number of characteristic pairs for the application of the numerical methods is about 60 [117].

It is interesting to note that the distinction between hierarchical and nested tree types of classifications explained in section 3.2.1 is made quite clear by these sponsors of numerical taxonomy. They probably make this distinction because the numerical methods result in a hierarchical classification that is not nested, whereas most other biological classifications have been nested trees.

A second classical problem in classification is the need to fulfill more than one purpose. Library Science is an outstanding example of a field that has such a problem. A classification for documents in a library normally needs to serve at least two primary functions: that of indexing for the retrieval of a specific document, and that of arranging the documents to allow the browsing through a sequence of documents on similar subjects. Both purposes require a classification using terms for the class names that indicate the content of the document. The multi-purpose aspect of library classification is amplified by the fact that libraries must be usable by people with a great variety of backgrounds. Often a class that is natural for one person is artificial for another.

A method that has found much favor among librarians in the past half century is that of faceted classification. Vickery defines facet analysis thus [130]:

> The technique of facet analysis is the conceptual analysis of a subject to pick out, from the theoretically unlimited number of characteristics by which a subject could be divided, those that are most significant in the study and practice of the subject.... Instead of first trying to construct from the original universe one vast tree of knowledge, facet analysis first groups the terms into categories...and then arranges the terms within each category into the form of a classificatory tree.

The essence of a faceted classification can be reduced to three features:

1) The classification consists of several more or less independent areas, called fields and facets by Vickery. A field can be thought of as a subject area (such as architecture) and a facet can be thought of as a way to classify a particular field (a classification of architectural objects might have facets for material, historical period, form, etc.).

2) Each facet is structured hierarchically and may have several levels.

3) Rules are provided for combining terms from different facets for the classification of a single document.

Vickery states that the criterion for the use of a given facet must be "literary warrant," which could be roughly translated as need, and he goes on to list a set of basic categories that he has found useful in developing facets. He cautions that this list is by no means conclusive, it is only a starting point and a guide. (See table 3.10 for his list; note that it is related somewhat to Aristotles "categories.") In a report on classification for the construction industry (in England), Vickery's categories are condensed to four: thing, attribute, operation and "place and time" [1].

Even the advocates of faceted classification point out that very general faceted classifications are quite difficult to construct. Some in library science are opposed to the new developments in faceted classification, in some instances for Metcalf's reason quoted in the previous section. The principal issue seems to be a debate over the goodness of enumerative classifications versus analytico-synthetic classifications. For the purposes of library science, an enumerative classification is one in which all possible classes are listed and assigned a unique notation, whereas analytico-synthetic classification leaves a great deal up to the discretion of the individual classing the document. In analytico-synthetic classification, the notation (the unique class for a particular document) is constructed from basic terms in the elementary facets and from connectors indicating how a subject of a document may be built up from several different terms. Metcalf objects to the large number of individual judgment decisions which must be made in the use of an analytico-synthetic classification [84].

Vickery's work on faceted classification has at least one more subject of interest to this study. That is, he proposes an order for combining terms from different facets, because these different terms must often be combined to represent a single subject. The order of terms that he suggests is shown in figure 3.3. This order is dependent on the "thing" as the central idea. This is somewhat different from Ranganathan's proposed ordering depending upon decreasing concreteness [106]. It is most closely related in the literature on classification to the proposal by Mills for utility [86].

3.2.5 Use of Classification in the Organizational Model

Clear logical principles exist for defining classes of objects. Although these time honored principles have proven to be lacking for certain tasks, and new principles have shown practicability in some instances, the overall goodness of a classification is still measured by its naturalness (or its fulfillment of intended purpose) and its ease of use.

Appropriate principles for grouping in organizing standards are a major issue in this study. The selection of a particular approach depends on a deeper consideration of the characteristics of provisions, which are the "objects" being classed. The issue of principles for grouping is returned to in section 4.2.

Vickery's suggested classes for the development of facets and the fundamental categories of Aristotle, Kant, and Pierce are of aid in developing suggested basic classes for building standards and of further aid in developing a classification for any specific standard.

Table 3.10 Vickery's Fundamental Categories (Reference 130)

Category	Example
Things, Substances, Entities	
Naturally Occurring	Minerals, Plants, Soils
Products	Bridges, Engines, Fibres
Mental Constructs	Equations, Rectangles
Their Parts	Beam, Wheel, Wing
Constituents, Materials	Metal, Glass, Nitrogen
Organs	Heart, Seed
System of Things	
Attributes of Things	
Qualities, Properties, including	Cohesion, Colour, Solubility
Structure	Layers, Profile
Measures	pH, Ampere
Processes, Behavior	Vibration, Inflammation
Object of Action (Patient)--Any Thing Can Be a Patient	
Relations Between Things, Interactions	
Effects	Inhibition, Stimulation
Reactions	Nitration, Symbiosis
Operations on Things	
Experimental	Cutting, Breeding
Mental	Calculation, Reasoning
Operating Agents--Any Thing Can Act as an Agent	
Properties of Attributes, Relations and Operations	Rate, Variation
Operations on Attributes, Relations and Operations	Measurement, Control
Place, Condition	
Time	

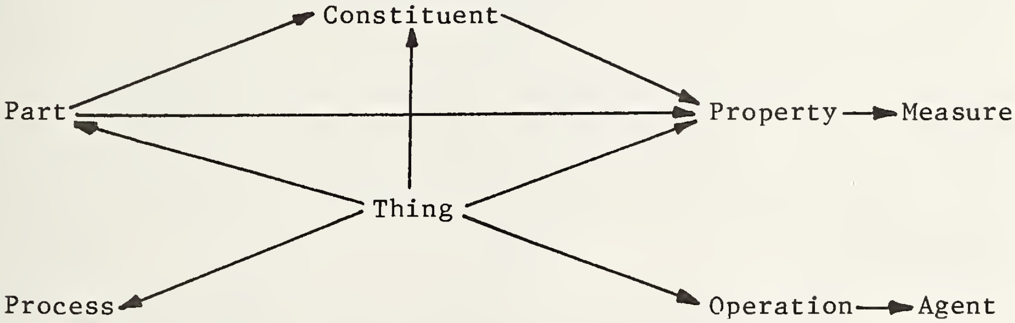


Figure 3.3 Vickery's Order of Terms (Reference 130)

3.3 Theories for the Structure of Provisions

Of all the case studies that classified sets of provisions, the one offering the most objective manner of developing relevant classification is that by Fenves, Rankin, and Tejuja [38]. In that study a common structure was identified for all types of provisions and standard "forms" were proposed for two types of performance provisions: performance requirements and performance criteria. These standard forms came from a grammatical analysis of the sentence structure for those provisions.

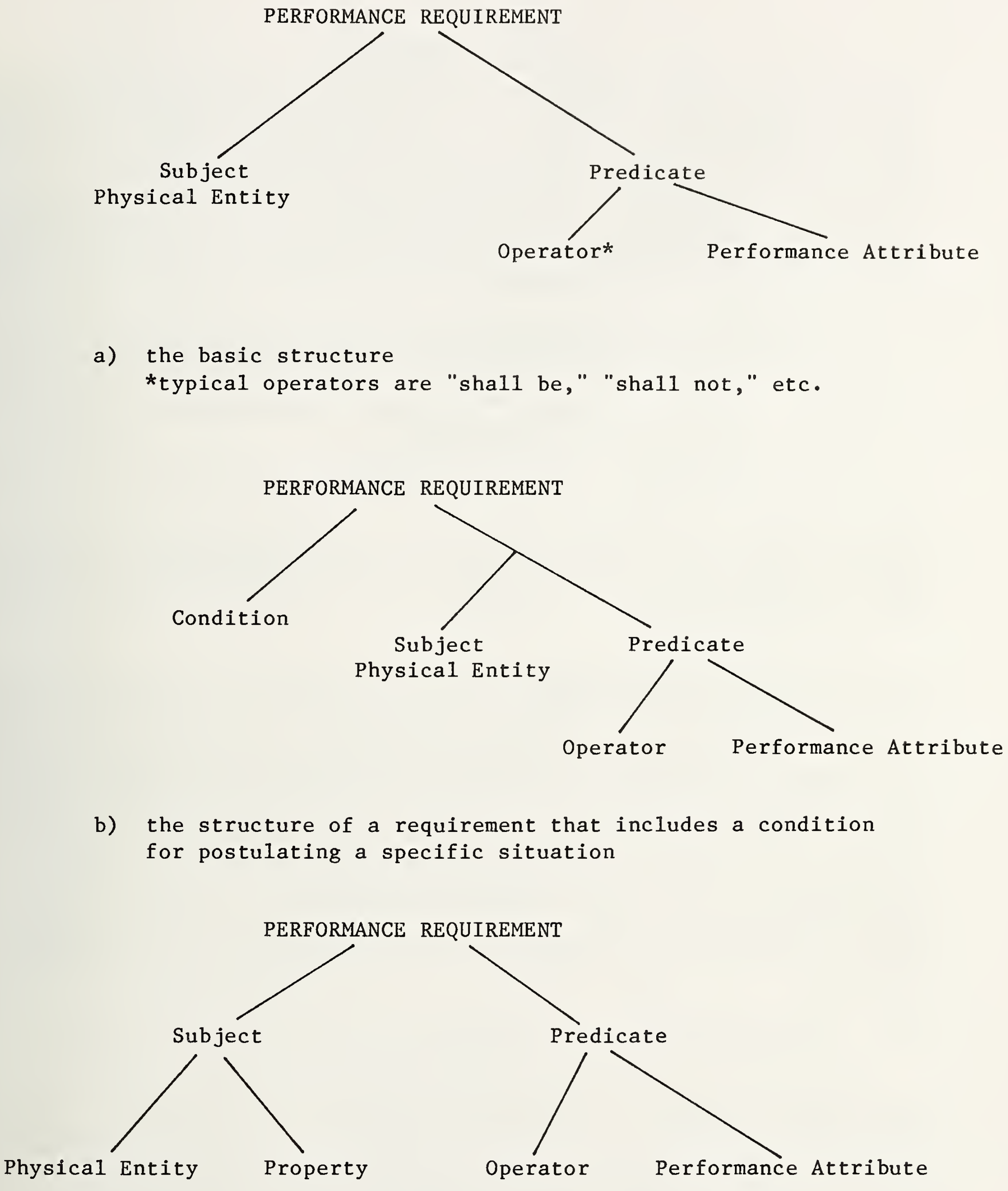
An aim of this study is to provide an objective basis for classifying provisions for all types of standards, a basis that will allow reproducible classing of provisions. Thus, a thorough review of that study is in order This also compels a review of the linguistic theories underlying the proposed standard forms.

3.3.1 Review of the Standard Forms Proposed by Fenves, Rankin, and Tejuja

The most basic concepts identified by Fenves, Rankin, and Tejuja are that each provision addresses a performance attribute of a physical entity and that all provisions have a two-part structure: the physical entity addressed by the provisions corresponds to the subject and the performance attribute addressed corresponds to the predicate. They state that this is not strictly true of the surface structure of the provisions they studied but that simple paraphrasing usually makes it possible to express the provision in such a manner. They then go on to propose standard forms for performance requirements and criteria but they did not study provisions for evaluation procedures in any detail. (Recall the RCEC format for performance standards, shown in figure 1.2.) Their forms are based on the structure they observed and are the basis of the guidelines for writing provisions that they later offer.

The standard form for a performance requirement is shown in figure 3.4a, with two options shown in figures 3.4b and 3.4c. They note that the first option, a condition, normally is used "for postulating a certain implementation" and that it is useful because it does not specify the implementation, it merely postulates what must be done should the implementation exist. The second possible modification is that of the property. They state that properties are used as modifiers of the physical entity in the subject position and usually follow the subject words immediately. As an example, they class the following performance requirement [59] in the three ways shown below (refer to the classification of table 3.5):

> System assemblies containing heat transfer fluids shall not leak to an extent greater than that specified in the design when operated at design conditions.
>
> - subject: "System"/"Transport"/"Mechanical"
> - predicate: "Operability"/"Dysfunctions"
> - property: "Exposure to or Contact with Fluid"



a) the basic structure
*typical operators are "shall be," "shall not," etc.

b) the structure of a requirement that includes a condition for postulating a specific situation

c) the structure of a requirement that defines the physical entity through the use of a property

Figure 3.4 Structure of Performance Requirements (Reference 38)

Figure 3.5 shows the standard form identified for performance criteria. Note that it differs from the form for performance requirements only in the predicate: the attribute is replaced by a measure, which is normally divided into two parts, a phenomenon and quantifier. This conforms with the definition of a criterion offered in section 1.4. Conditions and properties can be accommodated in the standard form of a criterion just as in a performance requirement, but they are not shown in figure 3.5. They class the following performance criterion [59] into three ways shown below (once again, see table 3.5):

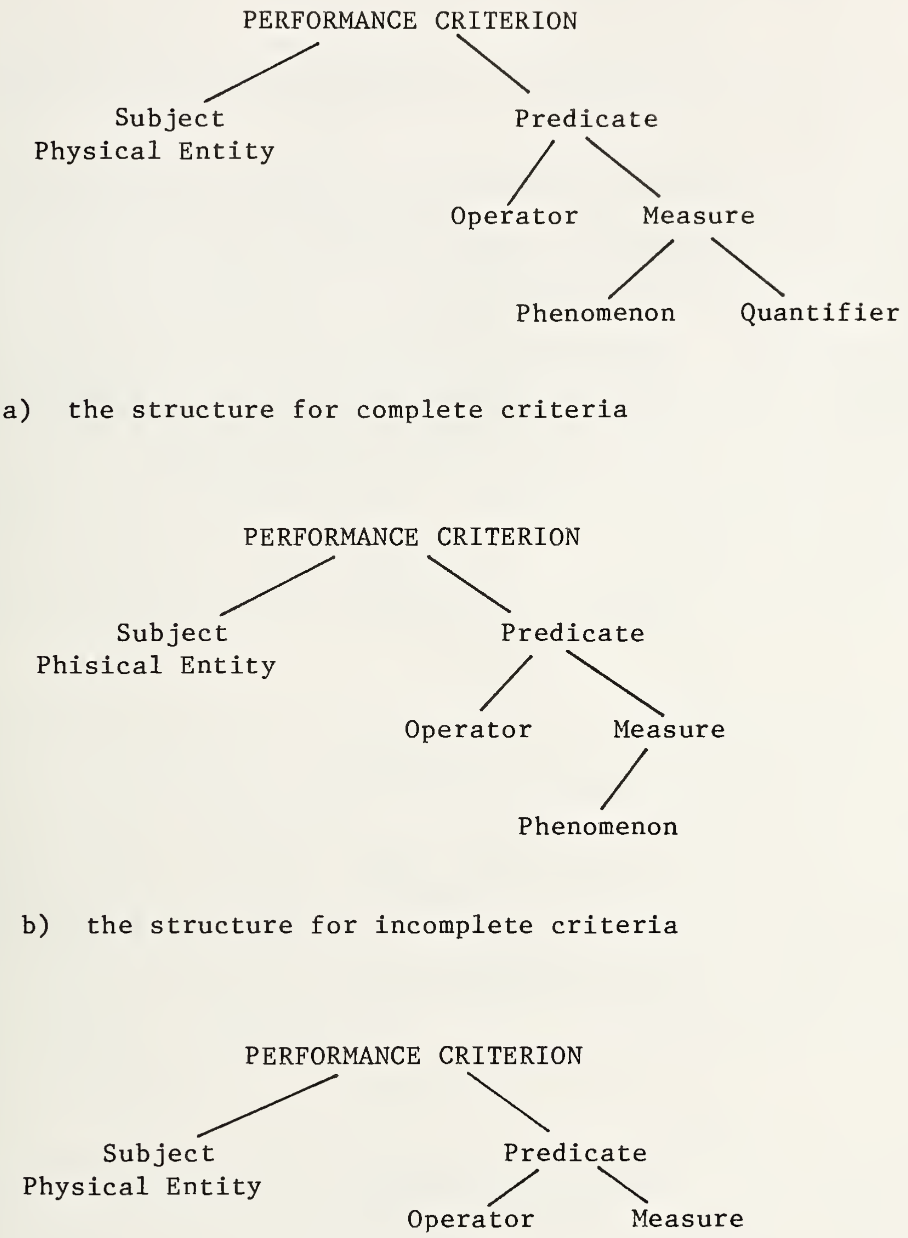
> Suitable connections shall be provided for the flushing (cleaning) of liquid energy transport systems.
>
> - subject: "System"/"Mechanical"/"Conveying"/"Piping"/ "Drain or Fill Attachments"
> - predicate: "Maintainability"
> - property: "Exposure to or Contact with Fluids"

Fenves, Rankin, and Tejuja also presented a classification of types of criteria based on the form of predicate actually observed in the criterion. This classification and its relation to the structures of figures 3.5 are shown in table 3.11.

Although their study concentrated on the structure underlying performance provisions, Fenves, Rankin and Tejuja recommend classifying all provisions by physical entity, performance attribute, and property. They point out that classification by property is advantageous for standards of the performance type because it allows the writer to avoid specific enumeration of physical entities, which tends to lead to prescriptive specifications. Before this work on performance standards can be generalized to apply to all types of standards it must be verified, in a sense, by further consideration of the paraphrasing mentioned earlier. Relevant classification is not likely if changing the form of a provision also changes its meaning. This is addressed by examination of the standard forms of provisions in the light of current theories on the syntax of sentences.

3.3.2 Linguistic Theories on the Relation of Meaning and Expression

The standard forms for provisions proposed by Fenves, Rankin and Tejuja [38] are based on a grammatical analysis of the sentences: that is, parsing a sentence into subject and predicate. Grammatical theories are part of larger overall theories in linguistics relating meaning to expression--or in the words of the Swiss linguist Ferdinand De Saussure [78]: "the relation between the concept and the acoustic image". Grammatical theories, indeed linguistic theories, abound. Some of the theories are controversial, both in their content and in the methodological aspects of their development. One reason that there are so many theories in this area is that significant effort in the automatic translation of our natural languages by computer was mounted in this country and elsewhere



a) the structure for complete criteria

b) the structure for incomplete criteria

c) the structure for implicit criteria

Figure 3.5 Structure of Performance Criteria (Reference 38)

Table 3.11 Types of Performance Criteria (Reference 38)

Type	Figure 3.5	Characteristic of Predicate
Implicit	c)	Reference to an external document
Explicit		
Incomplete	b)	No quantifier is stated
Complete		
Existence	a)	Existence is mandated
Value		
Document	a)	Quantifier value given in the standard
Design	a)	Quantifier value to be derived in design
Other Source	a)	Quantifier value to be obtained from external standard

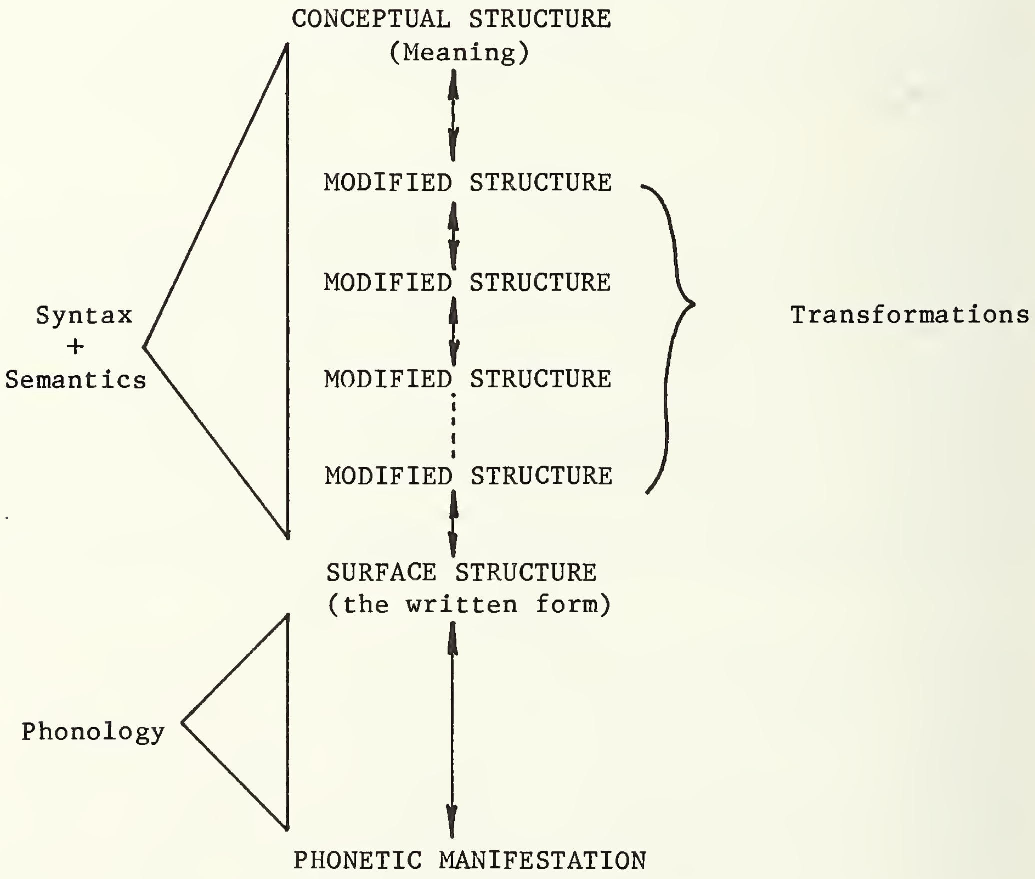


Figure 3.6 Langacker's Model of Linguistic Organization (Reference 73)

following the second World War. Bar-Hillel relates a significant portion of the history of machine translation in reference [10].

It is possible to extract several common features among the most significant theories. Each is a multi-level hierarchic structure of several theories, in which the various levels often tend to overlap and the boundaries frequently are not clear. Nonetheless, most theories include the following three levels: phonology, grammar and semantics [75]. Phonology deals with the acoustic production of words and sentences, thus it will be of little interest in this study. Grammar frequently is subdivided into syntax and lexicon. Syntax is the relation between words in sentences (actually, between word-like elements somewhat smaller than common words that consistently carry a single meaning). The lexicon is the list of these word-like units with their meanings and syntactical characteristics. Semantics is the study of the meaning of sentences and is not defined in a structural manner to the same extent that phonology and grammar are. In particular the relation between grammar and semantics is not well defined. Syntax and its relation to semantics are of interest in this study.

Langacker, a modern American linguist, gives a description of the theoretical structure of grammar that is designed to be less controversial than most by including concepts that are most widely accepted. As such, his description is useful to begin this study of the forms of provisions. In the following paragraphs a few important concepts from his book are summarized [73]. Figure 3.6 shows that syntax and semantics work together in relating the conceptual structure of a sentence to the surface structure of the sentence. Furthermore it shows that the relation between meaning and expression involves several transformations that produce modified structures. These modified structures and the transformations between them are an important concept, worthy of some explanation here, because the modifications are analogous to the paraphrasing referred to by Fenves, Rankin, and Tejuja.

One example of a relation between two modifications of the same structure is the active to passive transformation. The sentences "John hit the ball," and "The ball was hit by John," have the same meaning and thus the same conceptual structure. A different kind of relation between the surface and conceptual structures is shown by the ambigious sentence "Tom and Dick or Harry went to town," which is an example of one surface structure having two underlying conceptual structures. Synonymous sentences are sentences with different surface structures that may have the same conceptual structural such as "John went with Mary," and "John accompanied Mary." The transformation between conceptual structure and surface structure depends on a great number of things, and Langacker indicates that semantics is involved as well as syntax. An interesting point advocated by many European linguists is that the context and function of the sentence in the overall discourse is quite important in determining which transformations are observed between conceptual and surface structures [75 corroborated in 26]. Obviously, paraphrasing is a task to be undertaken only with great care.

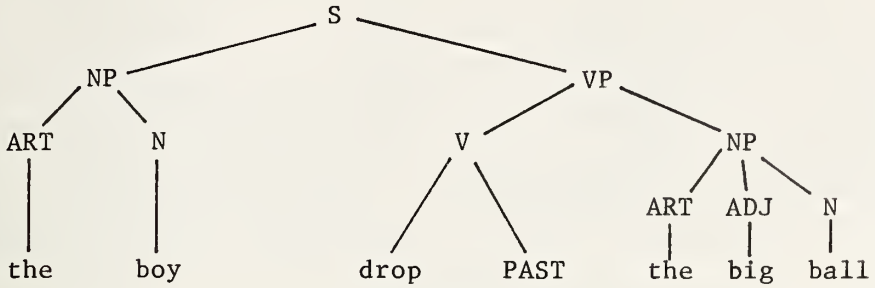
The second important concept that Langacker describes is the exposure of the surface structure of the sentence through constituent analysis. Bach, another linguist, points out that constituent analysis is just a modern name for the older concept of parsing [9]. Figure 3.7 shows the structure resulting from constituent analysis of the very simple sentence, "The boy dropped the ball." Note that the word dropped is divided into two constituents: the stem of the verb "drop" and the past tense portion of the verb. This is an example of the word-like units referred to earlier that are smaller than words. One important feature of the structure represented in figure 3.7 is that it is a tree. Also note the similarity of that structure with the standard forms proposed by Fenves, Rankin and Tejuja.

The tree shows that the initial noun phrase in a sentence is the subject and the verb phrase is the predicate. The object of the verb is part of the verb phrase, thus maintaining a binary structure of the tree. Some authors contend that all English sentences tend towards a binary tree structure [138], although this is by no means a feature found in the work of all authors. More complex sentences are typically formed in two ways: joining equal parts through the use of conjunctions such as and, or, but, etc., and embedding sentences or portions of sentences within sentences. Figure 3.8 shows the structure of a more complex sentence that involves both of these aspects. The sentence is "John walked but Mary rode the bike that Sue gave her." Note that a sentence may be embedded as a portion of a noun phrase.

Langacker also presents some examples of the structural transformations between underlying structures and surface structures [73]. As an example consider the figure 3.9, which contains an analysis of the phrase "the red barn," which might be the subject of a simple sentence. The figure shows that the phrase "the barn that is red" underlies the surface phrase "the red barn," and is in turn underlaid by the phrase "the barn the barn is red." The point of this example is that although adjectives often are adjacent to nouns as part of a noun phrase in the surface structure, the common underlying structure for all adjectives is that of a predicate adjective connected to the subject noun by a copula (a form of the verb to be). This fact becomes useful in developing a basic structure for a classification system.

Several authors describe linguistic theories on the relation between meaning and expression in more detail than Langacker. None is universally accepted, but Noam Chomsky's description of the theory known as transformational generative grammar is the most widely discussed structure for analyzing these relations [21, 22, 23, 71]. There is considerable debate on the complete validity of the theory, which indicates caution must be observed in any application [76]. His theories offer more insight on the subject of paraphrasing, and for the purposes of this study the similarities with other competing theories and the completeness of the theory are more important than the points of controversy.

Chomsky divides grammar (he uses the term differently than previously defined) into three parts: syntax, semantics, and phonology. He



Symbols:

S	Sentence
NP	Noun phrase
VP	Verb phrase
ART	Article
N	Noun
V	Verb
ADJ	Adjective
PAST	The past tense portion of a verb

Figure 3.7 Structure of a Simple Sentence

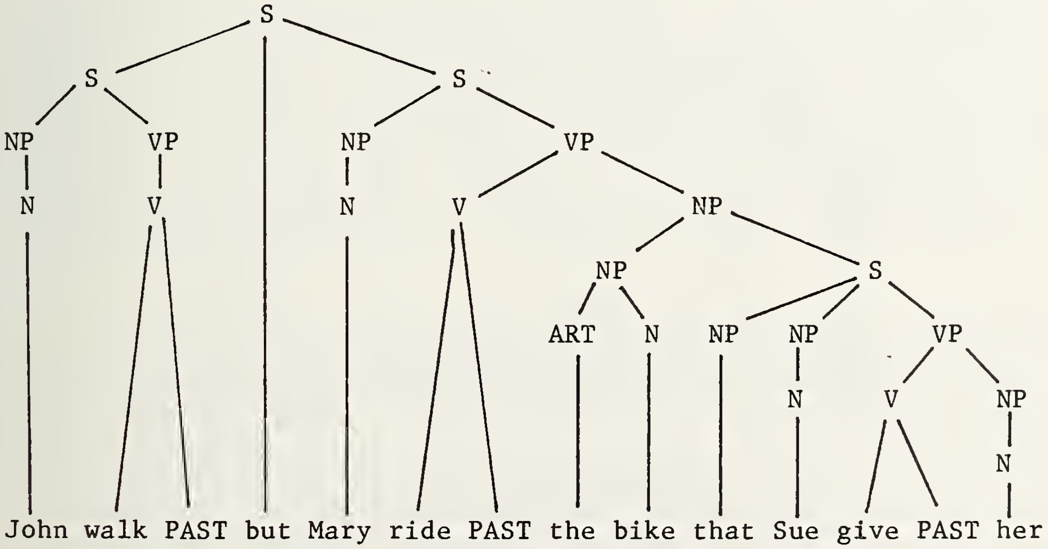
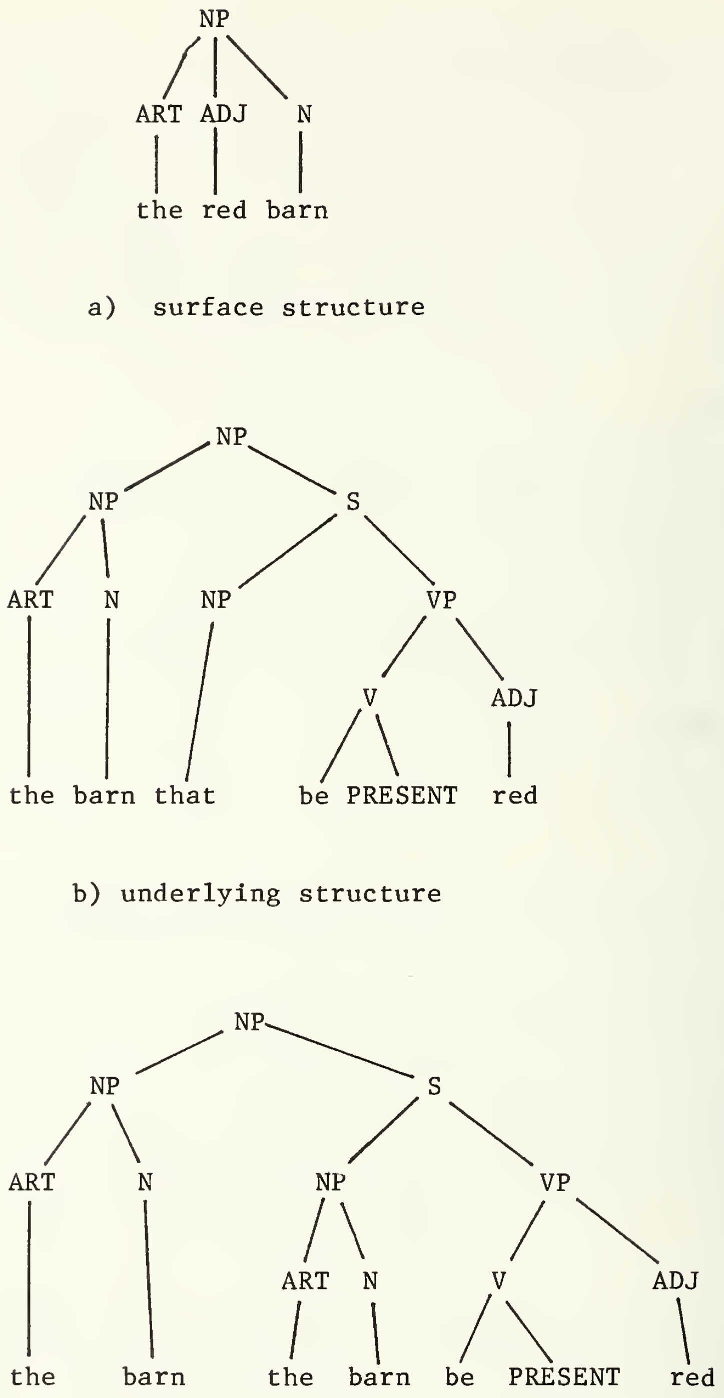


Figure 3.8 Structure of a Complex Sentence (Reference 73)



a) surface structure

b) underlying structure

c) deeper underlying structure

Figure 3.9 Example of Underlying Structure

concentrates entirely on syntax, which is the specification of the structure of a sentence. He defines semantics as the relation between the syntactic structure of a sentence and the semantic representation, and he defines phonology as the relation between the syntactic structure of a sentence and the phonetic signal. The basic element of his theory is that the structures operated on by semantics and phonology are not identical. That is, phonology operates on the surface structure and semantics operates on the deep structure. It is possible, but not common, for the deep structure and the surface structure to be identical.

Chomsky's syntax is divided into two components: the base component assigns (or "generates") the structural description to the deep structure of a sentence and the transformational component relates the structural description of the deep structure to the structural description of the surface structure. The portion of the theory that is controversial is the base component of syntax. In particular, many linguists doubt that a single structure identifiable as a deep structure exists, although there seems to be agreement that semantics operates on a different level than the surface structure of a sentence [75, 76]. In contending that his theories are not completely new and radical, Chomsky states that linguists of earlier centuries had ideas that were very similar to his [23].

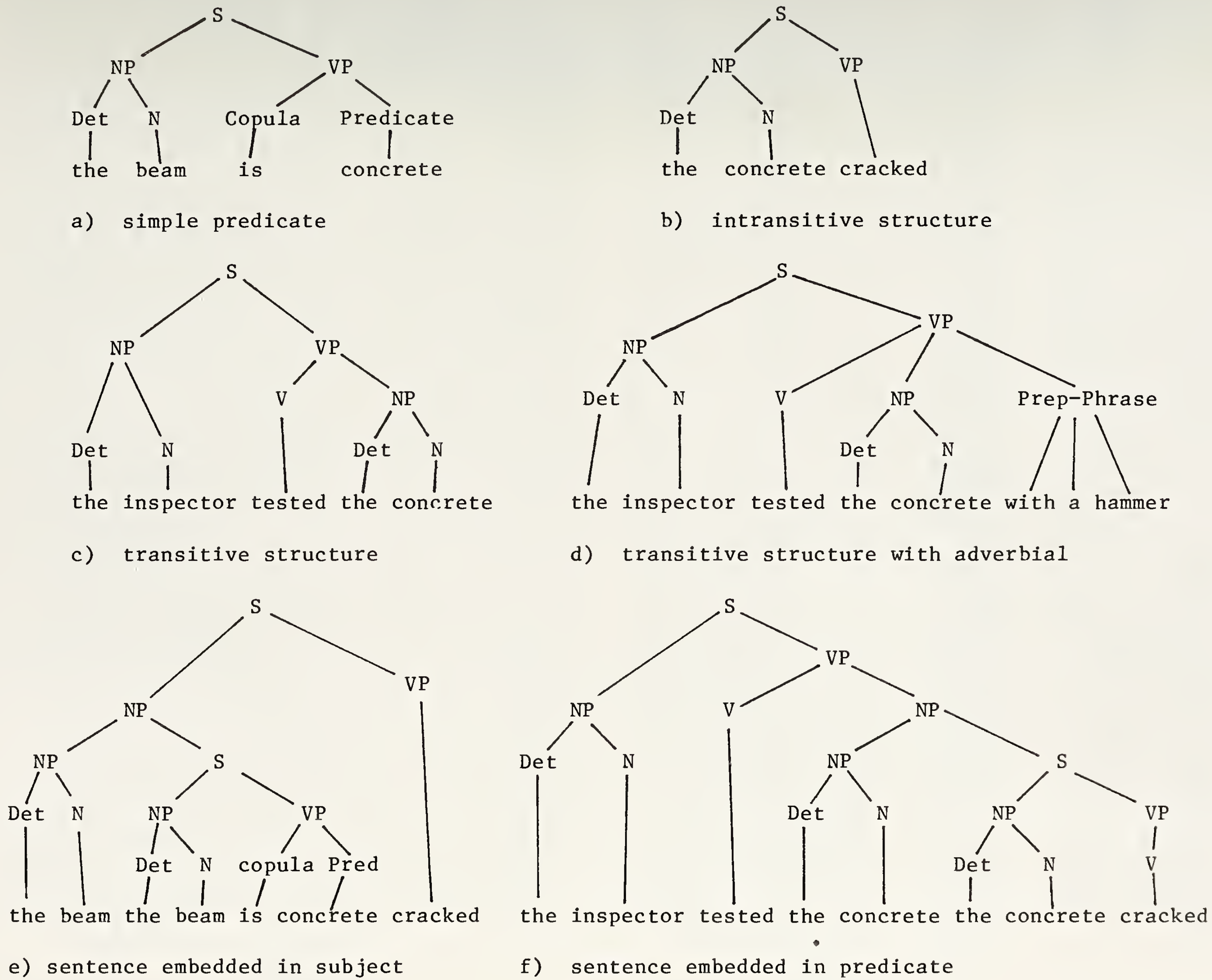
Chomsky divided the base component of syntax into two major portions. The first portion, which he calls his phrase structure rules, is a set of rules that govern the types of tree-like structural representations for sentences at the deep structure. The second portion is the lexicon that contains a list of all the words and word-like units together with the distinctive features that govern their use in the deep structure. Both portions offer some interest in this study.

Table 3.12 contains a representative set of phrase structure rules that are derived from Chomsky's work [21, 22]. For any node in the structural tree, these rules define the permissible constituent nodes at the next level of the tree. Figure 3.10 shows several examples of structural descriptions of sentences which can be derived from the phrase structure rules in table 3.12. Note that structural descriptions in figure 3.10 are quite similar to the structural descriptions offered for the surface structure of sentences earlier in this section, with a few exceptions.

For the purpose of this study it is worth noting that at the deep structural level sentences incorporate series of fragments that are typically declarative and quite simple. Chomsky calls these the basic strings, and for sentences with only one such basic string he uses the term kernel sentence for the basic string [21]. In this report, kernel sentence is used as the name for any simple declarative sentence contained within the deep structure. Other items in the description of the deep structure of interest to this study are that the "logical" subject of a sentence is the subject shown in the deep structure, that the logical subject is frequently not the subject shown in the surface structure, that the modification of nouns by adjectives typically reverts to a predicate modification in the deep structure, and that sentences of great length can be developed through the embedding of sentences.

Table 3.12 Phrase Structure Rules Derived from Chomsky (References 21, 22)

Parent		Children
S	→	NP + VP (+ Place) (+ Time)
NP	→	(Det) + N
NP	→	NP + S
VP	→	Copula + Predicate
VP	→	V + (+ NP) (+ Prep-Phrase) (+ Prep-Phrase) (+ Manner)
VP	→	V + S
VP	→	V + Predicate
V	→	(Aux) + Verb
Predicate	→	Adjective
Predicate	→	(like) Predicate-Nominal
Prep-Phrase	→	Direction, Duration, Place, Frequency, etc.


S
NP
VP
Det
N
Copula
Predicate
the
beam
is
concrete
S
NP
VP
Det
N
the
concrete
cracked
S
NP
VP
Det
N
V
NP
Det
N
the inspector tested the concrete
S
VP
NP
Det
N
V
NP
Prep-Phrase
Det
N
the inspector tested the concrete with a hammer
S
NP
VP
NP
S
Det
N
NP
VP
Det
N
copula
Pred
the beam the beam is concrete cracked
S
VP
NP
V
NP
Det
N
NP
S
Det
N
NP
VP
Det
N
V
the inspector tested the concrete the concrete cracked

a) simple predicate

b) intransitive structure

c) transitive structure

d) transitive structure with adverbial

e) sentence embedded in subject

f) sentence embedded in predicate

Figure 3.10 Samples of Possible Structures

The purpose of this examination of grammatical theory is to investigate the possibility of providing relevance in classification of provisions through paraphrasing provisions to a standard form. Chomsky contends that understanding a sentence is reduced (in a sense) to understanding the basic strings. In other words, the semantic component of language operates primarily on the deep structure. This is of interest in this study in spite of the previously cited fact that the most controversial portion of Chomsky's theory is the deep structure. Thus, the grammatical issues in this study can be characterized as follows:

1) A structural description of a sentence has been proposed as a basis for relevant classification of provisions (the standard form of Fenves, Rankin, and Tejuja). Does this correspond to either the surface structure or to any underlying structure such as kernel sentence?

2) Does the concept of kernel sentences provide a relevant and reliable basis for the classification of provisions?

3) If they are needed, can the kernel sentences representing the underlying structure of provisions easily be determined for the purpose of classing?

The first step is to re-examine the work done by Fenves, Rankin, and Tejuja in light of the grammatical theories reviewed here. Non-performance type provisions are examined in chapter 4.

One additional feature of interest in Chomsky's work that can be examined separately is the portion of the lexicon he called syntactic features. Syntactic features are ways of categorizing the words and the subword units contained in the lexicon. He offers the following five binary categories that may help in establishing categories in a classification of provisions, much as the categories of the philosophers mentioned in section 3.2.3: common, count, animate, human, and abstract. He points out that these five categories are in fact interrelated to some extent. Obviously human is inapplicable if the category animate is negative [21].

3.3.3 Examination of the Standard Forms in Light of Grammatical Theories

Fenves, Rankin and Tejuja state that some paraphrasing was necessary to modify the given expression of requirements and criteria to match the standard forms proposed in their report [38]. Obviously then, the surface structure of the provisions they studied does not always match the basic form they proposed. It would be useful to know how often the surface structure does match the standard form. For those cases in which the two do not match, it would also be useful to know if any underlying structure matches the standard form. A very brief and rapid study was made of the performance requirements studied by Fenves, Rankin and Tejuja for the correlation of the basic form with the surface and deep structure (or at least, what is believed to be the deep structure) of the original provision. Four categories are possible as follows:

1) both the surface and the deep structure correspond with the standard form

2) only the surface structure corresponds with the standard form

3) only the deep structure corresponds with the standard form

4) neither the surface nor the deep structure correspond with the standard form

Examples will be presented for three of these four categories. No provision was found for which the basic form did not match the surface structure but did match what might be the deep structure (case 3 above), even though such a situation was expected because it would result from logical paraphrasing.

The comparison can be visualized most easily by looking only at the subjects of the three different structures. Table 3.13 contains examples of each of the three categories found. At first glance it might seem that no consistent correlations can be drawn. However, a more detailed examination of the underlying structure of some of the sentences does shed additional light on the situation. Considering the third and last of the provisions contained in table 3.13, the following five kernel sentences can be identified as potentially being part of the deep structure:

s1) (?) shall make provisions

s2) provisions shall maintain quality

s3) heat transfer fluid has quality ("heat transfer fluid" is considered as one noun here)

s4) quality has level

s5) level does not impair function

These five kernel sentences are embedded in one another to produce a single sentence at the surface. Many of the provisions studied by Fenves, Rankin and Tejuja contain structures similar to the first kernel sentence identified here, namely a deleted passive actor, yet a human entity is never the subject of the standard form. Furthermore, in almost all of these provisions, the subject of the standard form does show up as the logical subject of <u>one</u> of the kernel sentences embedded in the underlying structure, heat transfer fluid in this instance. One of the recommendations Fenves, Rankin and Tejuja made concerning good style of writing provisions is that provisions should be written in an active, not a passive, sense [38]. Following this rule would avoid the occurrence of an unspecified actor appearing as the logical subject of the kernel sentence.

Table 3.13 Examples Comparing the Standard Form Proposed by Fenves, Rankin, and Tejuja with Surface and Underlying Structures

Provision	Category*	Subject of the:		
		Proposed Standard Form	Surface Structure	Underlying Structure
The control subsystem shall provide safe and efficient operation of the DHW, HC, and H systems.	1	control subsystem	control subsystem	control subsystem
The auxiliary energy subsystem shall be integrated into the H, HC, and DHW systems to the extent necessary to automatically provide the designed heating, cooling, and domestic hot water.	2	auxiliary energy subsystem	auxiliary energy subsystem	unnamed entity integrating the system
Provision shall be made to maintain the quality of the heat transfer fluid at a level that does not impair its heat transfer function.	4	transport fluid	provision	unnamed entity making the provision

* The categories are as follows:

1) both the surface and the deep structure correspond with the standard form

2) only the surface structure corresponds with the standard form

3) only the deep structure corresponds with the standard form (no example found)

4) neither the surface or the deep structure correspond with the standard form.

3.3.4 Application of Linguistic Theories to the Classification of Provisions

It is apparent that if the rule of writing active provisions is observed, both the surface structure and deep structure will correspond fairly well with the standard form proposed by Fenves, Rankin and Tejuja, at least for performance provisions. It also is apparent that the standard form is quite convenient for classing. At this point, it is not apparent whether consideration of underlying structures is necessary or practical in the classification of provisions in general. This will be explored further in the next chapter.

3.4 Issues for Improvement

The foregoing discussions of past case studies for organization, of classification, and of linguistics have served to identify the following subjects as the primary issues to be focused upon in this study:

1) Interface: the interface of the organization system with the remainder of the model for standard must be clear. The relation between individual datums and individual provisions must be established for the range of types of standards from performance to prescriptive and for the various types of provisions, thus defining the scope of the organizational system. The interface required between a particular outline and the information network must also be clearly defined.

2) Objective basis for relevant classification: the principles of classification for performance provisions established by Fenves, Rankin, and Tejuja need to be tested critically for potential generalization to other types of provisions with due account made for the linguistic underpinnings of those principles. This appears to be the key step in attaining reproducibility in the model.

3) Coordinated principles for classification: this important issue can actually be subdivided as follows:

   i) Overall structure of the classification: the apparent choice is between strictly logical systems, faceted systems, and numerically based systems.

   ii) The extent to which logical rigor is necessary: for dependable and unambiguous use, the most detailed levels of a classification need to satisfy the logical principles, and this need is absolute for automated processing of provisions. The extent to which the rigor is required must be determined with due attention paid to the overall structure of the classification.

iii) Detailed rules of classing provisions: for indexing and outlining purposes the principles of classing a given provision need to be defined and potential conflicts resolved.

4) Basic categories for building standards: the most relevant and meaningful categories for provisions found in building standards need to be identified so that the development of a classification will have a consistent starting point.

5) Detailed rules for the formulation of a new standard: the techniques for structuring a classification to predict necessary provisions must be explored and the principles defined.

6) Procedures for the development of indexes: since indexes are an important organizational tool and since they have not been a part of the model for standards, procedures for their development must be tested.

7) Procedures for development of outlines: a unified set of algorithms is needed that would be applicable for both top-down and bottom-up models of use. The procedures also need to incorporate the best available measures of quality for a specific outline.

The criterion that will be applied when choices are necessary in the introduction of new features into the model is that of simplicity. Classically speaking, this is Occam's Razor ("avoid unnecessary multiplication of entities" [131]), which is interpreted to mean that the simplest of competing theories is preferred to the more complex. That is, the method or principle that allows the model to accomplish all its purposes in the simplest overall manner is the one to select. The remainder of this report is devoted nearly entirely to the exploration of these issues, broken down in the following fashion:

1) Interface - sections 4.1.1 and 6.3

2) Relevant Basis - sections 4.1.1 through 4.1.4

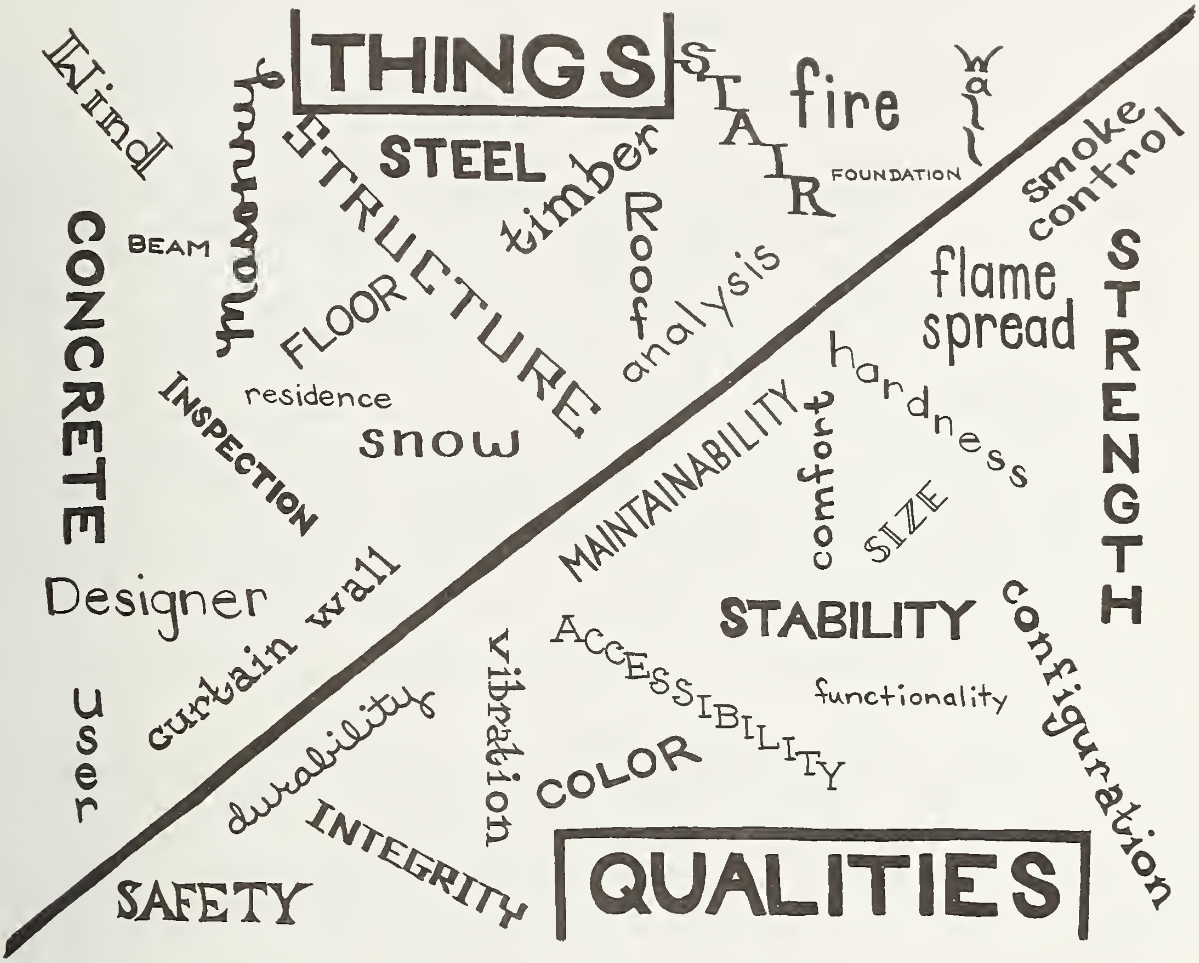
3) Principles for Classification - section 4.2

4) Basic Categories - section 4.3

5) Formulation - chapter 5

6) Indexing - section 6.1

7) Outlining - section 6.2



CHAPTER 4

DEVELOPMENT OF THE CLASSIFICATION FOR A STANDARD

In this chapter, various factors in a model framework for classifying provisions are presented and evaluated. The several objectives for organization that were identified in the earlier chapters are used as objectives for the classification system. The principal concerns of the chapter are:

1) to formulate a consistent interface of the organizational system with the decision table and information network system

2) to define the basis for relevant and meaningful classification of provisions

3) to formulate rules for the structure of a classification and the classing of provisions

4) to develop the basic framework of a classification applicable to design standards in the building area

Each of these concerns about a classification system is treated in the light of the eventual purpose of the classification for working with the scope and the arrangement of the standard. The first section of the chapter deals with the first two concerns just listed while the second and third sections of the chapter deal respectively with the third and fourth concerns. A sample classification is presented and discussed in the fourth section.

4.1 Characteristics of Provisions

The purpose of the classification system is to distinguish one provision from another. This is the third and major sense of classification as explained in section 3.2, namely distinguishing one object from another. However, it must be recalled that the objects being classified are actually mental constructs; the tangible manifestation of the provision (its written form) does not contain significant characteristics for classifying. The number and type of words and letters are not of any importance. The meaning of the provision is what provides the important characteristics for classification.

To begin, it is easy to divide the characteristics of a provision into two parts: what the provision does in the standard (for example, tests compliance or develops parameters) and what the provision addresses (the subject and qualities considered). A study of what a provision does allows the definition of a consistent interface with the decision table and information of network systems. This aspect of classification of provisions is discussed in section 4.1.1. A study of what the provision addresses is carried on in the subsequent subsections and is, in fact, an extension of the type of classification of performance provisions carried out by Fenves, Rankin and Tejuja and discussed in chapter 3.

4.1.1 Functional Types of Provisions

In chapter 1 a very general definition of a provision was offered, that is, that a provision is effectively a stipulation. It was also noted that requirements and criteria are important types of provisions. At this point it is proper to further refine the definition of a provision, and the definition offered is that a provision is a rule for judging compliance with a standard. Note that a decision table is a collection of rules. However, no one-to-one correspondence between provisions and decision tables has yet been established. Furthermore since one decision table corresponds to one datum, it is apparent that no one-to-one correspondence between a provision and a datum has yet been established.

In chapter 1 the first special type of provision defined was a requirement, and it was noted that a requirement is normally characterized by a value of satisfied or violated. Examination of previous studies involving decision table representation and analysis of standards indicates that there is a correspondence between requirements and datums, to wit, each requirement corresponds to one datum. The value of the requirement is the value of the datum. Furthermore, since a requirement cannot be an input value for a standard, each requirement is a derived node, usually a decision table. In terms of the correspondence between provisions and datums there is little difference between a requirement and criterion. A criterion also carries a value of satisfied or violated. A criterion is simply a special type of requirement, one that has an explicit test associated with it. Therefore there is the correspondence between criteria and datums that is identical to the correspondence between requirements and datums.

Examination of previous work with decision tables reveals that not all derived datums are requirements or criteria; there are many datums for which the value is not "satisfied" or "violated." Performance theory is of some aid in examining this issue further. It is generally accepted that there are three essential elements to a complete performance statement: a requirement, a criterion and a test method or evaluation procedure [14, 132]. Examination of various performance standards that are written with this format reveals that the test methods or evaluation procedures are essentially provisions for the determination of the values compared in the criteria; that is, they are ingredients to the criterion.

This examination of previous decision table studies for standards that range from performance to prescriptive reveals that all the derived nodes that are not requirements are similar enough to be considered one class, called "determinations." Thus it is possible to group all derived datums into two classes: requirements and determinations. It is a simple extension of these ideas to group all provisions into the same two classes. (Hereafter initial capitalization is used for Requirement and Determination to remind of this basic separation.) This allows the desired correspondence between datums and provisions. In this model a criterion is treated synonymously with a Requirement.

To illustrate these two classes, several previous studies using decision tables to represent the information in a standard are now examined. In much of the early work (reference 36 is a good example) one level of "terminal criteria" are defined and several levels of "working tables" are ingredients to these terminal criteria. Each of the terminal criteria are Requirements and each of the working tables are Determinations. In more recent work using decision tables (reference 25 is a good example), this rigorous structure is not observed, however, placing all derived datums into the two categories is still a natural dichotomy. An important observation about these later studies that is consistent with the earlier studies is that the output nodes of the information network are always Requirements. One further observation, which is pertinent because it is distinctly different from the earlier studies, is that Requirements are frequently found several levels into the information network from the output levels.

It must be mentioned that the work by Noland and others (reference 92 is a good example) is not directly comparable. Effectively, all the decision tables presented in those studies are of the kind called "switching" tables in the description in chapter 2. That is, the tables are used only for the purpose of determining the applicability of various provisions. Typically, the action sets of those decision tables are statements of requirement type provisions. Noland used this explicitly different format in his studies to provide a test for the relative merits of each type of presentation [92].

A case can be made that this dichotomy observed among decision tables in past studies is no coincidence, nor is it simply an effect of the techniques used in formulating decision tables. The dichotomy also seems to be natural when examining the textual expression of provisions taken at random. For example, consider the following provisions taken from the Uniform Building Code [126]:

> Allowable Floor Areas
>
> Sec. 505.(a) One-Story Areas. The area of a one-story building shall not exceed the limits set forth in Table No. 5-C except as provided in Section 506, nor the limits specified in Chapter 16.
>
> For buildings located in Fire Zone No. 3, the basic area may be increased by 33-1/3 percent.
>
> (b) Areas of Buildings Over One Story. The total area of all floors of multistory buildings shall not exceed twice the area allowed for one-story buildings. No single floor area shall exceed that permitted for one-story buildings . . .
>
> (d) Area Separation Walls. Each portion of a building separated by one or more area separation walls may be considered a separate building provided the area separation walls meet the following requirements . . .

The first sentence can be read as two provisions. The clause "the area of a one-story building shall not exceed the limits" is a Requirement which may either be satisfied or violated. The remainder of the sentence can be looked upon as a provision which defines how the limits used in the prior requirement are to be determined, that is, a Determination. The second quoted sentence is simply a rule for establishing a value, the basic area, which is used in other Requirements. Thus, it is a Determination.

The third and fourth sentences are both Requirements. Note that both of them specify a physical entity and place a limit of some kind on that physical entity. The fifth provision is interesting, because it is a Determination, but the value is not explicity stated. The value that is determined by the provision is the actual area used for comparison with the allowable area derived in other provisions. An interesting point is that this actual area is dependent upon the value of another Requirement.

This particular portion of the Uniform Building Code has been analyzed and represented as a set of decision tables [49]. The six decision tables representing the six provisions (five sentences) just discussed are shown in tables 4.1 through 4.6. Note that tables 4.2, 4.4, and 4.5 contain information that is not in the quoted provisions. For example, table 4.2 contains the provisions in section 506 which were referenced from first

Table 4.1 Decision Table for Single Story Area Requirement

			1	2
1	Actual Area of Story ≤ Total Allowable Story Area	*	Y	N
	**			
1	Single Story Area Requirement = Satisfied	*	X	
2	Single Story Area Requirement = Violated	*		X

Table 4.2 Decision Table for Total Allowable Story Area

			1	2	3	4	5	6	7	8	9
1	Allowable Area From Chapter 16 = Unlimited	*	N	Y	Y	Y	Y	Y	Y	Y	Y
2	Occupancy Group = B4	*	.	N	Y	Y	Y	-	-	-	-
3	Construction Name = Open skeleton frame	*	.	.	N	Y	N	.	.	.	.
4	Fire Resistance of Entire Building < 1 hour	*	.	.	N	.	Y	.	.	.	.
5	Occupancy Group = D1	*	.	N	-	-	-	Y	Y	-	-
6	Number of Stories = 1 and Construction Type = III - 1 hour or IV - 1 hour or V - 1 hour	*	.	.	.	.	.	N	Y	.	.
7	Occupancy Group = C	*	.	N	-	-	-	-	-	Y	Y
8	Actual Maximum Exit Distance ≤ 50% of Allowable Exit Distance From 3302	*	.	.	.	.	.	.	.	N	Y
	**										
1	Total Allowable Story Area = Allowable Area From Chapter 16 * Area Increase for Fire Zone	*	X								
2	Total Allowable Story Area = Allowable Area From Table 5C * Allowable Area Multiplier From Section 506 * Area Increase for Fire Zone	*		X	X			X		X	
3	Total Allowable Story Area = Unlimited	*				X					
4	Total Allowable Story Area = 400 ft^2 * Area Increase for Fire Zone	*					X				
5	Total Allowable Story Area = 3900 ft^2 * Area Increase for Fire Zone	*							X		
6	Total Allowable Story Area = 1.5 * Allowable Area from Table 5C * Allowable Area Multiplier from Section 506 * Area Increase for Fire Zone	*									X

Table 4.3 Decision Table for Area Increase for Fire Zone

			1	2
1	Fire Zone = 3	*	N	Y
	**			
1	Area Increase for Fire Zone = 1	*		X
2	Area Increase for Fire Zone = 1.33	*	X	

Table 4.4 Decision Table for Total Building Area Requirement

		1	2	3	4	5	6
1	Actual Area of Total Building ≦ 2* Total Allowable Story Area	Y	N	N	N	N	N
2	Basement or Cellar Present = True	.	Y	Y	Y	Y	N
3	Basement or Cellar Qualifies as a Story = True	.	N	N	N	Y	.
4	Actual Area of Total Building - Actual Area of Basement or Cellar ≦ 2* Total Allowable Story Area	.	Y	Y	N	.	.
5	Actual Area of Basement or Cellar ≦ Total Allowable Story Area	.	Y	N	.	.	.
1	Total Building Area Requirement = Satisfied	X	X				
2	Total Building Area Requirement = Violated			X	X	X	X

Table 4.5 Decision Table for Area Requirement

		1	2	3	4	5	6	7	8
1	Single Story Area Requirement = Satisfied for all stories	Y	Y	Y	Y	Y	N	N	N
2	Number of Stories > 1	N	Y	Y	Y	Y	.	.	.
3	Total Building Area Requirement = Satisfied	.	Y	Y	Y	N	.	.	.
4	Occupancy Group = H	.	N	Y	Y	.	.	.	.
5	Occupancy Name = Open parking garage	.	.	.	.	.	N	Y	Y
6	Requirement for Area of H Occupancy = Satisfied	.	.	Y	N	.	.	.	.
7	Area Requirement for Open Parking Garage = Satisfied	.	.	.	.	.	.	Y	N
1	Area Requirement = Satisfied	X	X	X				X	
2	Area Requirement = Violated				X	X	X		X

Table 4.6 Decision Table for Actual Area of Story

		1	2
1	Area Separation Wall Requirement = Satisfied	Y	N
1	Actual Area of Story = Area of Story Across Entire Building		X
2	Actual Area of Story = Area of Story Within Area Separation Walls	X	

sentence of section 505. The added material in tables 4.4 and 4.5 also comes from closely related material that is not apparent from the provisions quoted here.

Table 4.7 presents several illustrative examples of Requirements and Determinations taken from other previous studies involving decision table representation of standards. No difficulty was encountered in classing datums or provisions from any of these studies.

Thus the proposed bifurcation of types of provisions and types of derived datums seems reasonable for the interface of the organizational system with the decision table and information network system. Individual provisions that are Requirements will be identified with a unique datum, thus the arrangement of Requirements can be independent of the information network.

The correspondence between Determinations and datums is not the same, however. Several Determination type provisions may produce values for a single datum, and thus be represented by that one datum. For example, the provisions from which rules 5 and 7 of table 4.2 were taken (those leading to actions 4 and 5) are found in chapters completely separate from the one excerpted here.

Therefore the arrangement of Determinations should not be independent of the information network. Locating the various provisions that govern the determination of the same datum in different places will be equivalent to splitting a decision table. In normal situations the placement of all provisions dealing with the determination of a single datum at one position in a standard seems to be the simplest possible solution. In those situations where this is not possible or not desirable, it is possible to provide cross-references between the disjoint locations.

This mismatch between provisions and datums when dealing with Determinations is not as serious as it might first appear. The current method of organizing performance standards is to let the arrangement of the performance requirements and performance criteria completely govern the organization. The evaluation procedures are located at each criterion, thus the organization of the requirements specifies the overall organization of the standard. It is but a small generalization of this concept to make the following claim. A standard is a set of rules that must be evaluated to determine compliance with the purpose of the standard. The evaluation of a rule may depend upon other rules in a recursive fashion. The overall organization of the standard can be completely described in terms of the organization of those rules that are the Requirements, since the value from a rule that is a Determination is never an output value of the standard.

A related issue that might seem somewhat tangential to this organizational study is the completeness necessary for adequate representation of a standard in a decision table format. The issue is related to the division of provisions and the subject is important to the overall use of decision table, network, and classification technology for the

Table 4.7 Samples of Requirement and Determination Provisions

Text of Provision	Source	Reference Study	Datum Name	Type*
"Buildings or other structures and all parts thereof shall be designed and constructed to support safely all loads, including dead loads, without exceeding the allowable stresses . . ."	[15]	[25]	"Safe Support Acceptable"	R
"The live loads to be assumed in the design of buildings and other structures shall be the greatest loads that probably will be produced by the intended use or occupancy, but in no case less than the minimum uniformly distributed unit loads required by Table 1."	[15]	[25]	"Required Live Loads-Floors"	D
"The effective slenderness ratio Kl/r shall not exceed 200."	[119]	[97]	"Compression Member Check"	R
"$\phi_c = 0.86$ for $\lambda \leq 0.16$ $\phi_c = 0.95 - 0.25 \lambda$ for $0.16 \leq \lambda \leq 1.0$ $\phi_c = 0.65$ for $\lambda \geq 1.0$"	[43]	[98]	"Determination of ϕ_c"	D
"The minimum thicknesses stipulated in Table 9.5(a) shall apply for one-way construction unless the computation of deflection indicates that lesser thicknesses may be used without adverse effects."	[16]	[92]	Not Applicable	R
". . . deflections shall be computed taking the modulus of elasticity for concrete as specified in Section 8.3.1 for normal weight or lightweight concrete and taking the effective moment of inertia as follows . . ."	[16]	[92]	Not Applicable	D

* R = Requirement
D = Determination

representation of standards. Furthermore it leads naturally to an examination of the structure of Requirements. Therefore a short discussion is in order.

As an illustrative example consider the following provision taken from the new tentative provisions for seismic design of buildings [7]:

> Every building shall be designed to resist the total lateral seismic force determined in accordance with the following formula: $V = C_s D$ ...

A brief analysis of the underlying structure (as described in section 3.3.2) for this provision reveals three kernel sentences:

1) (?) design building

2) building resists force

3) formula determines force

It thus appears that this single sentence contains three separate provisions, the first two of which appear to be Requirements. In fact only one of these three provisions shows up in a recent decision table representation of the new seismic provisions [50]. The one that does is the third one, the Determination. The first of the three provisions is not included, because in the context of the overall tentative standard it is not necessary for determining compliance with the standard. The second Requirement is not included because it is redundant, although it is not possible to deduce that from simply reading this particular sentence. In fact extremely similar Requirements are found throughout the document, one of which is accompanied by a considerably more specific provision that is obviously intended to be used as the measure of the requirement. That provision is:

> Individual members shall be sized for shears, axial forces, and moments determined in accordance with these provisions, and connections shall develop the strength of the connected members or the forces indicated above.

For the purpose of judging compliance with the standard, it is not necessary to satisfy the redundant Requirement. Thus, neither is it necessary to insert a requirement datum in the formal expression that the building resist the force at the point that the sentence "Every building shall . . ." would have it inserted. Frequently, the textual expression of provisions and standards leads to redundant statement of Requirements in an absolute sense. It is not so common that the textual expression leads to redundant statements of provisions that determine values used in judging the requirements. The formal representation need not be so exhaustive that it include all redundant statements. Indeed, the formal representation should discourage textual redundancy.

Given that the Requirements are a subset of the provisions sufficient to define the organization, it is appropriate at this point to turn away from the study of functional types of provisions and examine the other aspects of Requirements which might provide a relevant basis for classification.

4.1.2 Structure of Requirements

A Requirement is a stipulation, or a rule, that can be characterized as "satisfied" or "violated." Consideration of what a Requirement addresses leads naturally to two categories: a generalized "thing" and a generalized quality required of that thing. These categories are parallel to the first fundamental distinction upon which classification is based, that is between an object and its attributes. At this point THING should not be thought of only as a tangible object; systems, activities, and other more abstract notions may be included. According to Webster, thing is ". . . whatever exists, or is conceived to exist, as a separate entity, or as an individual quality, fact, or idea . . ."[131]. Similarly, REQUIRED QUALITY includes concepts like existence of a THING and relations between THINGS in addition to more conventional characteristics of a THING. These two categories are adequate to represent an idealized structure and content of Requirements for the purposes of classification for organization, as described in this section.

Note that the two categories for a Requirement differ only slightly from the statement of Fenves, Rankin and Tejuja that [38]:

> . . . provisions in all types of specifications contain the same type of information, namely: a <u>subject</u>, which refers to or makes some mention of a physical entity, and a <u>predicate</u>, which relates to the attribute to be satisfied.

First the categorization of information at this point is applied only to requirements, not to all types of provisions. Second, the concept of THING is somewhat broader than the concept of physical entity, in that THING can easily include more abstract notions. Third, REQUIRED QUALITY is somewhat different in concept than attribute, which Fenves, Rankin and Tejuja linked specifically to performance attributes.

Fenves, Rankin and Tejuja proposed three major independent classifications for provisions: by physical entity, by property, and by performance attribute. However, the example they provide for classification by property is quite similar to their own example of classification by physical entity. The classes of "Mechanical" physical entities in table 3.5 (such as "Motive," "Heat Exchanger," "Conveying," and "Hanger") do not appear to be fundamentally different than the classes of properties under "Exposure or Contact with Fluids" in the same table (such as "Potable Liquids," "Corrosive Liquids," "Pressurized Liquids" and so on). Likewise, neither of the classes under "Conventional Structural Assemblies" ("Ultimate Strength" and "Working Stress") in the physical entity classification in table 3.5 is significantly different than the class "Requiring Cutting," which is under "Structural" in the property classification. In both classifications the classes are used to identify or describe physical entities through the use of qualities.

In fact it is quite difficult to make a clear distinction between the name of an object and the function of an object. (Kant [62]: "... we know not this thing as it is in itself but only know its appearances ...") Thus if a classification of physical entities is based on names of objects, and a class of properties is based on qualities of objects, it is difficult to separate them consistently. For example consider the word pump. It can be used as a name of an object, but it is also the name of the function of the object. Likewise the word manifold is commonly taken to be the name of an object, but the word clearly implies something about the shape, which is a property of the object.

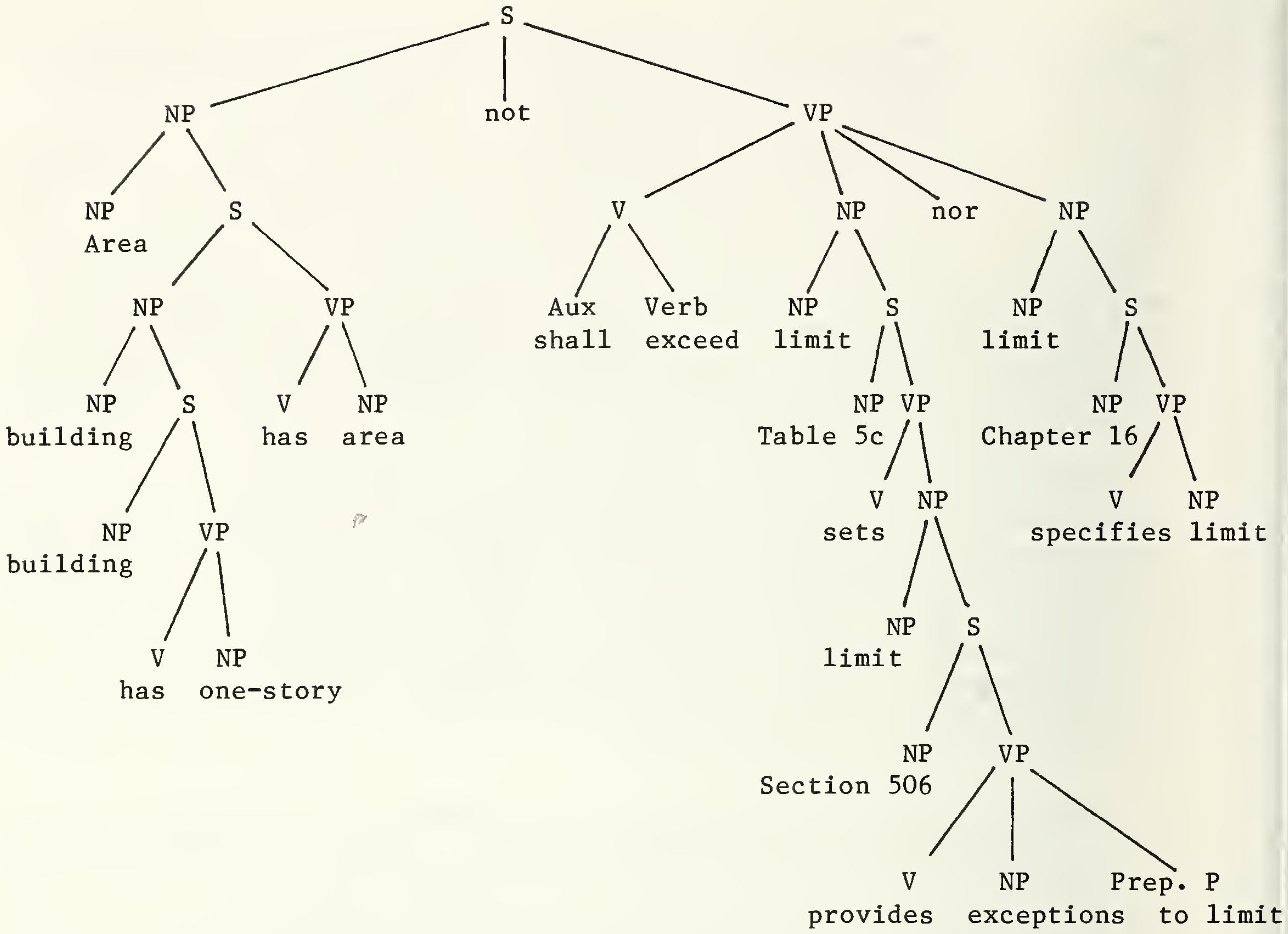
The classification of physical entity and the classification of property forwarded as examples by Fenves, Rankin and Tejuja can be seen in the light of Vickery's works on classification as simply being two facets of the classification for the subject of their performance requirements. (It is useful to recall at this point that there is really little difference between the modification of a noun by a leading adjective, by following relative clause, or by following predicate, as was shown in figure 3.9 for the word red as it modifies "the red barn". See reference 100 for a fuller discussion of the meaning of words and other signs.)

What is offered thus far is a rather slight modification of the structure of Requirements in performance standards. The real question is whether the structure of Requirements, that is the categorization of their information, is applicable in other types of provisions beyond performance standards. Investigation of potential extensions leads to a few observations. First, the THING, or the subject of provisions, in general cannot be limited strictly to physical entities, although this is by far the largest and most significant class of subjects. Second, the REQUIRED QUALITIES are often quite difficult to relate to performance attributes. But most important, the two part categorization of the information does seem to stand up.

Consider the Requirements among those provisions from the Uniform Building Code quoted in the preceding section as an example of requirements dealing purely with physical entities. It was pointed out that the sentence:

> The area of a one-story building shall not exceed the limits set forth in Table No. 5-C except as provided in section 506, nor the limits specified in chapter 16.

contains both a Requirement and a Determination. The THING addressed by the Requirement, a physical entity in this case, is a one-story building. The REQUIRED QUALITY is the area of the one-story building. Obviously the subject of the surface structure of the sentence is the REQUIRED QUALITY. Furthermore, as shown in figure 4.1, the REQUIRED QUALITY appears to remain in the subject position for the underlying structural description of the sentence. Far from being an unusal occurrence, the placement of the REQUIRED QUALITY in the subject of a Requirement is quite common. Indeed, both the



The area of a one-story building shall not exceed the limits set forth in Table No. 5-6 except as provided in Section 506, nor the limits specified in Chapter 16.

Figure 4.1 Potential Underlying Structure for the Single Story Area Requirement and Determination

other Requirements in the provisions quoted from the Uniform Building Code contain the REQUIRED QUALITY in the subject position for any potential structural description of the sentences.

Typically, it is easy to paraphrase a Requirement so that the REQUIRED QUALITY is in the predicate position, and the THING is in the subject position, however. Consider the following paraphrase as an example, "One-story buildings shall have an area less than the limits set forth ..." This is equivalent to taking the first kernal sentence embedded in the original subject, area, as the starting point for the new sentence. No claim is made that the paraphrased Requirement is identical to the original Requirement. None of the theory reviewed in section 3.3.2 would support such a claim. To attempt to do so would involve a far deeper study of semantics and its interaction with syntax than is being conducted here.

The claim being made is simply this: it is generally possible to paraphrase a Requirement into the form of a simple sentence with the THING as the subject and the REQUIRED QUALITY as the predicate, and this forms a relevant basis for classifying that Requirement. The nuances of word order for actual expression are beyond the scope of this study, although the simple and consistent sentence construction used as a tool for classification would lead to clear and easy-to-use standards should that same construction prevail in the actual writing of provisions. In most cases, it does not seem too difficult to discern which kernal sentence in a complex Requirement is most appropriate. Subsequent examples help illustrate this.

Although a physical entity is generally identifiable for all Requirements, there are many instances in which it must be inferred or in which it is so general as to be relatively useless for classing. In most of these instances there is a more relevant way to establish the subject and required quality for classing purposes. For example, consider the following requirements, all taken from the new seismic design provisions [7]:

1) Each contractor responsible for the construction of a Designated Seismic System, or component, listed in the quality assurance plan shall submit a written statement to the regulatory agency prior to the commencement of work on such system or component.

2) The regulatory agency may require the submission of a written report which shall include . . . the determination of lateral pressures on basement and retaining walls . . .

3) The building owner shall employ an approved special inspector to observe the construction of all Designated Seismic Systems in accordance with the following requirements . . .

4) The analysis shall include, . . . at least the lowest three modes of vibration or all modes of vibration with a period greater than 0.4 seconds, whichever is greater . . .

5) The required periods and mode shapes of the building in the direction under consideration shall be calculated by established methods of mechanics for the fixed base condition . . .

The one thing that each of these examples has in common is that it is difficult to identify a quality which would reasonably be required of a physical entity, even though it is usually possible to identify, or at least infer, a physical entity. Another thing that the Requirements have in common is that they all involve activities or processes that have some relation to human involvement. In fact, each of these Requirements could be classified with a human entity as a subject (the THING) and some quality required of the human entity as the predicate (usually the quality required is the conduct of an activity). Each also could be classified with a process as the subject (the THING) and a quality required of that process as the predicate (typical qualities required of processes would be their existence, a method to be used in their conduct, or documentation of the process, etc.). (Hereafter Physical Entity, Process, and Human Entity are shown with initial letters capitalized as an indication that the reference is to a THING being used for purposes of classification.)

Thus the THING for the first Requirement listed could either be the "contractor" or the general Process of "quality assurance." The THING for the second Requirement could be the implied Process of "soil investigation" or the unnamed Human Entity that must carry out that soil investigation, should it be required by the regulatory agency. Note that the Requirement could also be classed in a different fashion, with the "regulatory agency" as the THING and with the REQUIRED QUALITY being the action of specifying whether the soil report is required or not. The third Requirement contains two Human Entities, either one of which could be the subject of a Requirement. Likewise the Process of "quality assurance" could be the THING for a semantically related Requirement. In the fourth and fifth examples the Human Entity is unnamed, but is implied to be the "designer of a building."

The examples show that Human Entities for use as a THING in classification may be named or unnamed in the verbal or written expression of the Requirement. The examples also show that a Process which may be used for subject classification purposes may also be named or unnamed in the written expression of the Requirement (the Process "soil investigation" is definitely not named in example 2). In addition, for those Processes that are named in the written expression of the Requirement, the method of naming is variable. The Process may be named in the subject (such as example 4, "analysis") or may be the passive verb for a named or unnamed human actor (such as "calculated" in example 5).

It appears that selection of either Physical Entity, Process, or Human Entity as a THING for classing a Requirement can be relevant, and the one that fits the best should be selected. It can be assumed that there will be examples in which two or more will be useful for classification purposes. It is also possible to perceive that for some Requirements naturally occurring processes that involve no Human Entity would be the THING for classification purposes. A Requirement dealing with corrosion of a steel member, for example, might well be classified by the natural Process "corrosion" rather than by the Physical Entity.

A general guide for the most relevant THING for classing a Requirement that shows a complex relation among potential THINGS is found by considering the clearest identification of a REQUIRED QUALITY. Thus example 4 is more relevantly classed as the Process "analysis" rather than the Physical Entity "structural system", because the REQUIRED QUALITY dealing with the "extent" of the analysis would provide a more meaningful organization than the REQUIRED QUALITY of being "well analyzed."

Ideally the classification of REQUIRED QUALITIES should reflect an inherent basis in performance, for to paraphrase a report on the development of performance specifications [14], every prescriptive Requirement is based on an implicit performance Requirement. In practice it proves nearly impossible to base a classification of REQUIRED QUALITIES for prescriptive Requirements completely on the Performance Attributes. (Hereafter Performance Attribute is capitalized to show its relation to REQUIRED QUALITY.) Common Requirements, at least in the area of building standards, have frequently been in use for a great many years and the original performance-based reason for their use is lost in history [120]. In addition, a single prescriptive Requirement may in fact serve as a measure for many different performance Requirements. In some instances, the original reasoning for the adoption of prescriptive Requirements is rational while in others it is empirical, and there may be no way of sorting these reasons out. Knowledge of necessary theory is sometimes inadequate [79]. Finally, Requirements used over a great many years tend to become self-perpetuating in that the side effects they bring along become rational reasons for continuing their use. Thus, although the classification of Requirements by Performance Attributes is important, frequently such a classification cannot adequately classify all the REQUIRED QUALITIES of Requirements for the purposes of organization.

In summary then, a simple structure for Requirements is postulated for use in classification. That structure is as shown in figure 4.2. The condition is retained for the same reason given by Fenves, Rankin, and Tejuja, that is to postulate a particular implementation when a Requirement applies only in that circumstance. The THING (subject) may be a Physical Entity, a Process, or a Human Entity. Within one standard, provisions may relate to any or all categories of THING. The categories of THING and REQUIRED QUALITY that are useful for development of a classification are discussed in more detail in section 4.3.

In using such a structure, one must realize that it is an imperfect model, a paragon of what Requirements might be, not a representation of what they actually are. Nonetheless the model appears to have a sound rational basis for providing relevance, and empirical evidence suggests that it is meaningful enough to be useful.

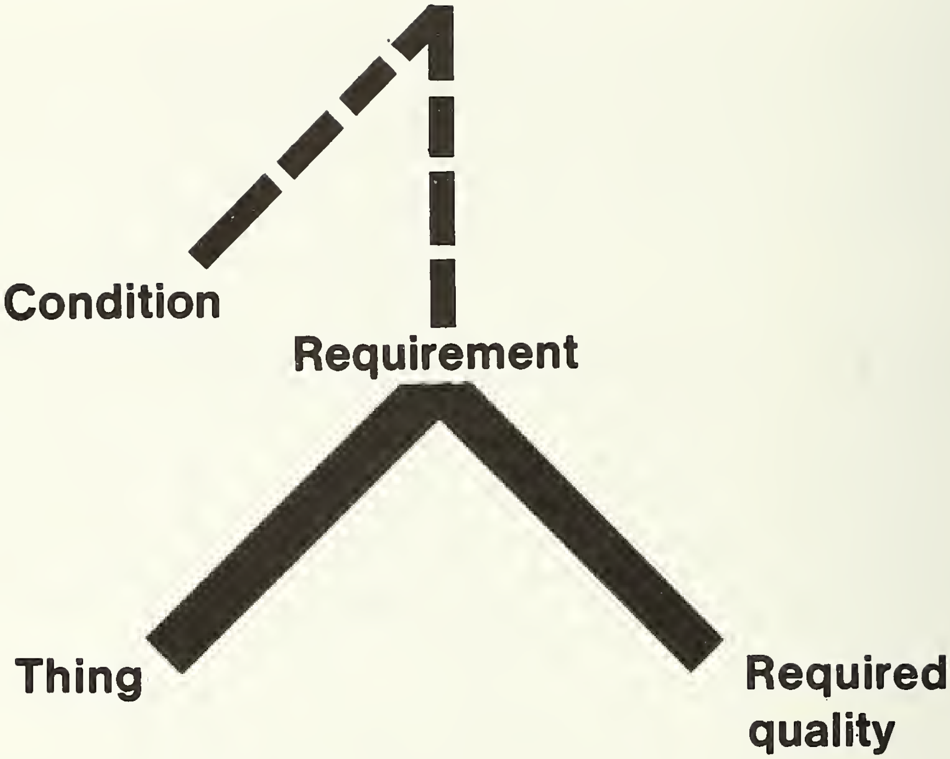


Figure 4.2 Model Structure of a Requirement

4.1.3 Types of Requirements

A study of the new seismic design provisions [7] and the formal documentation developed for them [50] reveals a useful taxonomy of Requirements. In light of the interface between the organizational system and the datum, decision table, and information network systems discussed in section 4.1.1, and the characteristic structure of Requirements identified in section 4.1.2, six types of Requirements may be seen. The distinction of the six types depends on:

1) singular structure of the Requirement

2) the original statement of a Requirement included in the conditions of the decision table

3) the presence of another Requirement among the ingredients of the Requirement

4) conditional applicability of an ingredient Requirement

5) the equivalence of the decision table as a whole to a new Basic Requirement

The types are defined as follows:

1) Basic Requirements have a singular subject and a singular predicate. They do not directly depend on other Requirements.

2) Multiple Requirements have plural subjects and/or predicates. They do not directly depend on other Requirements.

3) Cumulative Requirements depend unconditionally on other Requirement datums. The decision table does not contain an original statement of a Requirement.

4) Application Requirements depend conditionally on at least one of the ingredient Requirements. The decision table does not contain an original statement of a Requirement, nor is it equivalent to a new Basic Requirement.

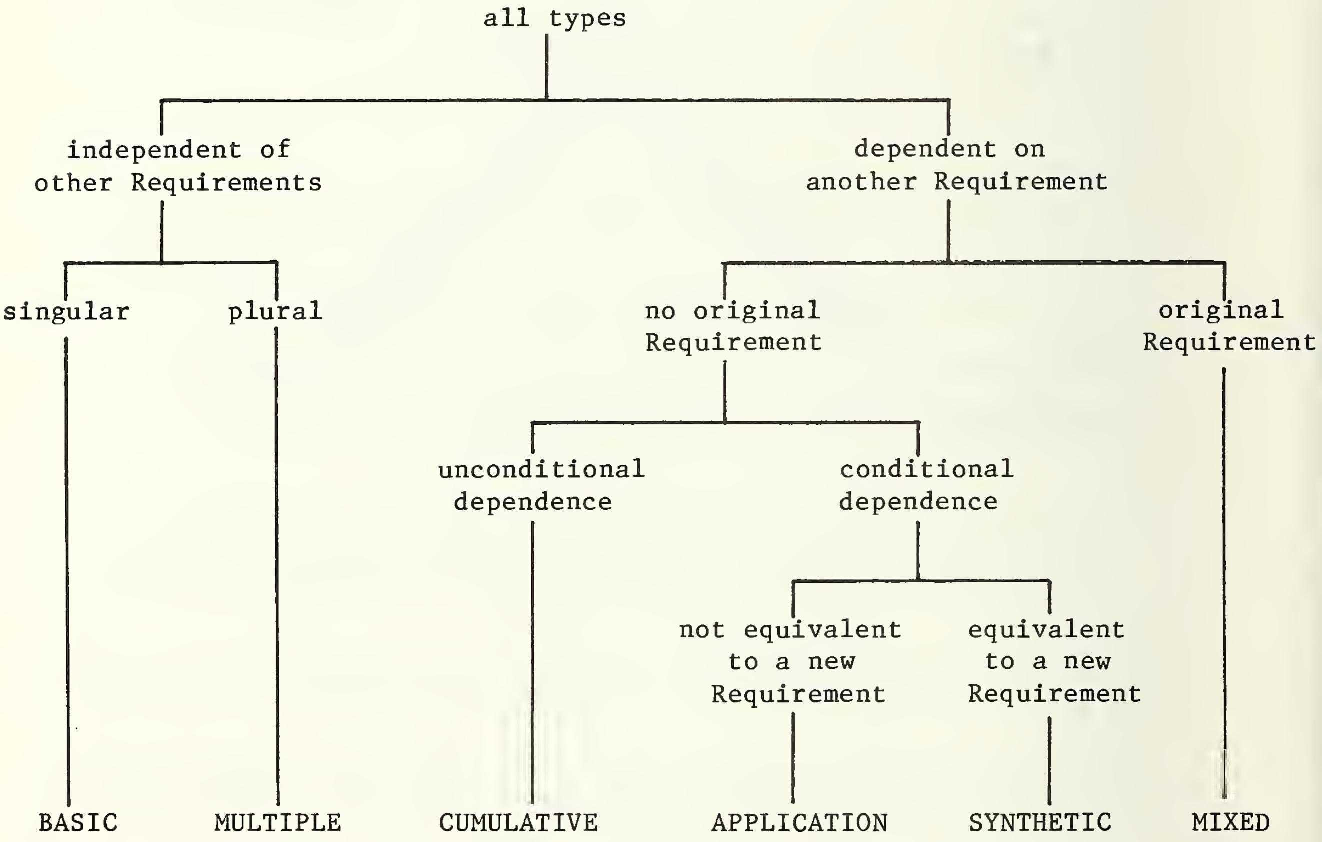
5) Synthetic Requirements are like Application Requirements except that they are equivalent to a new Basic Requirement.

6) Mixed Requirements depend directly on other Requirements (either conditionally or unconditionally) and the decision table does contain an original statement of a Requirement.

Each of these six types are illustrated with examples in the following paragraphs. Figure 4.3 shows a decision table and tree representing the definition of the six types.

		1	2	3	4	5	6
1	Direct dependence on other Requirement	N	N	Y	Y	Y	Y
2	Original statement of a Requirement among conditions	+	+	N	N	Y	Y
3	Singular subject and predicate	Y	N	.	.	.	.
4	Conditional dependence on a Requirement	.	.	N	Y	Y	.
5	Equivalent to a new Requirement	.	.	.	N	Y	.
1	Basic	X					
2	Multiple		X				
3	Cumulative			X			
4	Application				X		
5	Synthetic					X	
6	Mixed						X

a) decision table



b) decision tree

Figure 4.3 Definition of the Types of Requirements

As an example of a Basic Requirement, consider the following provision:

> The repair or alteration of an existing building subject to these provisions shall either (1) not reduce the lateral force resistance of the building below the requirements of these provisions or (2) shall provide for the seismic forces determined in accordance with these provisions . . . [7]

The decision table for this Requirement is shown in table 4.8. The REQUIRED QUALITY is clearly seismic force resistance, (or lateral force resistance--the reference source at times uses these terms interchangeably [50]), or, more simply for classification purposes, "strength". The THING is also singular, but it presents an interesting example of the interplay of Physical Entity and Process. In this case "building undergoing alteration or repair" is a more appropriate THING than the Process of "alteration" because the REQUIRED QUALITY pertains to the building, not the Process. Thus Process is a facet used for classification of a Physical Entity in this instance.

As an example of a Basic Requirement in which process is a more appropriate THING, consider:

> The internal forces in the members of the building shall be determined using a linearly elastic model. [7]

This Requirement could be classed as "analysis" for the THING and "modeling assumptions" for the REQUIRED QUALITY. The Physical Entity "structural members" could be used as a facet of the THING. ("Structural" is added to "member" because it is contained in prior heading for the section from which the Requirement was excerpted.)

Basic Requirements can involve a more complex statement than the preceding examples, yet still maintain the essence of a singular THING and REQUIRED QUALITY. Consider the following provisions and the corresponding decision table, shown in table 4.9:

> Buildings assigned to Seismic Hazard Exposure Group III shall be accessible during and after an earthquake. Where access is through another structure that structure shall conform to the requirements for Group III. Where access is within 10 feet of side property lines, protection against potential falling hazards from the adjacent property shall be provided. [7]

As stated in section 1.4, this is really three provisions, one general and two particular. It can easily be treated as a single Requirement, but could also be treated as three Requirements. Choosing the former way, the THING is a building assigned to "Seismic Hazard Exposure Group III" and the REQUIRED QUALITY is "accessibility". The first two conditions of the decision table contain the essence of the requirement. (However, the first condition easily could be removed from the decision table and used only in the organizational system.) The remainder of the paragraph and of the decision table are simply

Table 4.8 Sample Decision Table for a Basic Requirement

		1	2	3
1	Seismic force resistance before proposed activity ≤ Seismic force resistance after proposed activity	Y	N	N
2	Seismic force resistance after proposed activity ≥ Seismic force resistance required by these provisions	.	Y	N
	**			
1	Alteration and repair requirement = satisfied	X	X	
2	Alteration and repair requirement = violated			X

Table 4.9 Sample Decision Table for a Basic Requirement with Supplementary Special Provisions

		1	2	3	4	5	E
1	Seismic Hazard Exposure Group = III	N	Y	Y	Y	Y	
2	Building is accessible during and after earthquake = true	.	Y	Y	Y	Y	
3	Access provided by adjacent structure = true	.	N	Y	N	Y	
4	Seismic hazard exposure group of adjacent structure = III	.	.	Y	.	Y	
5	Distance from access point to side property line < 10 feet	.	N	N	Y	Y	
6	Protection provided against potential adjacent hazards = true	.	.	.	Y	Y	
	**						
1	Group III access requirement = satisfied	X	X	X	X	X	
2	Group III access requirement = violated						X

Table 4.10 Sample Decision Table for Requirement with two THINGS

		1	E
1	Building has capacity to function immediately after EQ = true	Y	
2	Designated systems have capacity to function immed after EQ = true	Y	
	**		
1	Group III functional requirement = satisfied	X	
2	Group III functional requirement = violated		X

extra measures necessary to satisfy the Requirement in certain special circumstances. Treating the paragraph as three Requirements would mean that the decision table would be a multiple Basic Requirement, which is the type discussed next, but in this instance the case for such treatment appears weak.

The simplest type of Multiple Requirement has either a multiple subject or a multiple predicate, as in the following (also see table 4.10):

> Essential facilities, and designated systems contained therein, shall have the capacity to function during and immediately after an earthquake. [7]

The two THINGS are "Group III buildings" and "designated systems in Group III buildings" ("Group III" is synonymous with "Essential Facility") while the single REQUIRED QUALITY is "functionality" at a particular time. Another example of a Multiple Requirement is (also see table 4.11):

> The capacity of the foundation soil in bearing or the capacity of the soil interface between pile, pier or caisson and the soil shall be sufficient to support the structure with all prescribed loads, without seismic forces, taking due account of the settlement that the structure can withstand. For the load combination including earthquake as specified in Sec. 3.7, the soil capacities must be sufficient to provide resistance at the elastic limit or less considering both the short time of loading and the dynamic properties of the soil. [7]

It is quite similar except that the THING can be considered singular, "soil", while the REQUIRED QUALITIES are plural: "load capacity", "stiffness", and "elastic limit", or possibly just "strength" and "stiffness".

In both the prior examples, the problem encountered in classification is that two or more classes intended to be mutually exclusive (for example, "strength" and "stiffness") apply to a single Requirement. Either the logical principles must be violated or the datum must be classed according to a more general, and thereby less relevant, class. Furthermore, the freedom to arrange the standard is compromised because all the Basic Requirements in the datum must remain together. A more complex Multiple Requirement is illustrated by the following (also see table 4.12):

> Each contractor responsible for the construction of a Designated Seismic System, or component, listed in the Quality Assurance Plan shall submit a written statement to the Regulatory Agency prior to the commencement of work on such system or component. The statement shall clearly show the following:
>
> 1. His acknowledgement that he is aware of the special requirements contained in the Quality Assurance Plan.

Table 4.11 Sample Decision Table for Requirement with Three REQUIRED QUALITIES

		1	E
1	Soil capacity under non-seismic conditions ≥ Required strength without seismic load and Settlement under non-seismic conditions ≤ Maximum settlement structure can withstand	Y	
2	Elastic limit of soil under seismic conditions ≥ Required strength	Y	
1	Foundation soil capacity requirement = satisfied	X	
2	Foundation soil capacity requirement = violated		X

Table 4.12 Sample Decision Table for a Complex Multiple Requirement

		1	E
1	Statement is written = true	Y	
2	Statement is submitted prior to start of work on DSS = true	Y	
3	Statement acknowledges awareness of reqts of Q A plan = true	Y	
4	Statement acknowledges that control will exercised = true	Y	
5	Statement contains procedures for control = true	Y	
6	Statement contains method, freq, and distr of reports = true	Y	
7	Statement names person responsible for control = true	Y	
8	Statement names position of person responsible for control = true	Y	
1	Contractors quality assurance statement requirement = satisfied	X	
2	Contractors quality assurance statement requirement = violated		X

2. His acknowledgement that he will exercise control to obtain conformance with the Design Documents approved by the Regulatory Agency.
3. His procedures for exercising control within his organization, the method and frequency of reporting and the distribution of the reports.
4. The person exercising such control and his position in the management of the organization. [7]

The problem encountered here is whether the THING is properly a Physical Entity (the "statement"), a Human Entity (the "contractor"), or a Process ("quality assurance"). A case can be made for each, but the case for the Process is quite strong because all the REQUIRED QUALITIES then can be characterized by a class such as "documentation" (of the Process).

Cumulative Requirements, as far as a decision table representation of a standard is concerned, are the result of the literal representation of catch phrases such as, ". . . all the requirements of this section," or ". . . all the following provisions," etc. Generally speaking, they are a reflection of the given organization of a standard in the decision table representation, which is not usually desirable because the freedom to rearrange the standard is again compromised. For example, the following provisions are a part of a chapter titled "WOOD".

> 9.4 SEISMIC PERFORMANCE CATEGORY B
> Buildings assigned to Category B shall conform to all of the requirements for Category A and to the additional requirements of this Section.
> 9.4.1 DETAILING REQUIREMENTS
> The construction shall comply with the requirements given below.
> (A) ANCHORAGE OF CONCRETE OR MASONRY WALLS. The diaphragm sheathing shall not be used for providing ties and splices required in Sec. 3.7.5 and 3.7.6.
> (B) LAG SCREWS. Washers shall be provided under the heads of lag screws that would otherwise bear on wood.
> (C) ECCENTRIC JOINTS. The 50 percent increase permitted for allowable working stresses in Sec. 208B of Ref. 9.1 shall be limited to joints where all parts of the joint are located within five times the depth of the member from the end of the piece. [7]

There are actually four Basic Requirements. The last phrase under the heading 9.4 and the sentence directly following the heading 9.4.1 can also be interpreted as a Requirement, as shown in table 4.13. When such a datum is included, the organizational system is constrained to locate within one section all the Requirements that are ingredients to the datum or else provide cross-references. Yet the Cumulative Requirement adds nothing to the information necessary for judging compliance with the standard that

Table 4.13 Sample Decision Table for a Cumulative Requirement Grouped by a Derived Class

			1	E
		*		
1	Category A wood requirement = satisfied	*	Y	
2	Category B wood tie requirement = satisfied	*	Y	
3	Category B lag screw washer requirement = satisfied	*	Y	
4	Category B eccentric joint requirement = satisfied	*	Y	
	**	*	**	**
1	Category B wood requirement = satisfied	*	X	
2	Category B wood requirement = violated	*		X

Table 4.14 Sample Decision Table for a Cumulative Requirement Grouped by Type of Object

			1	E
		*		
1	Ordinary concrete flexural member reinforcement requirement = satisfied	*	Y	
2	Ordinary concrete flexural member moment resistance requirement = satisfied	*	Y	
3	Ordinary concrete flexural member reinforcement anchorage = satisfied	*	Y	
4	Ordinary concrete flexural member web reinforcement requirement = satisifed	*	Y	
	**	*	**	**
1	Ordinary concrete flexural member requirement = satisfied	*	X	
2	Ordinary concrete flexural member requirement = violated	*		X

cannot be provided by the organizational system--that is the applicability of Requirements for a clearly defined category of subjects.

As a further example, consider the following provision in a chapter titled "REINFORCED CONCRETE":

> 11.6 <u>REQUIREMENTS FOR ORDINARY MOMENT FRAMES ASSIGNED TO CATEGORY B</u>
> Ordinary moment frames shall . . .
>
> > 11.6.1 FLEXURAL MEMBERS
> > Flexural members shall be provided with . . . reinforcement in conformance with the requirement given below . . . [7]

Table 4.14 shows a Cumulative Requirement that corresponds precisely to the section 11.6.1. Inclusion of it effectively constrains the organizational system from exercising the option to group provisions for longitudinal reinforcement in various members at one location and provisions for web reinforcement at another location.

Cumulative Requirements typically have a broad scope as compared to their ingredient Requirements, and they must be classed accordingly. The example from table 4.14 must therefore lose the classifier "reinforcement" since one of its ingredients does not deal with reinforcement (condition 2).

Application Requirements may be a simple reflection of the grouping of Requirements, such as Cumulative Requirements, or they may reflect more complex statements of the scope and applicability of certain Requirements. Table 4.15 is an example of the former type. It is taken from the following expression:

> The design and detailing of components of the seismic resisting system and of other structural and nonstructural components shall be as specified in this Section. [7]

plus the headings of each of the subdivisions of the section. Each of the four Requirements referenced in conditions 5 through 8 is one of the subdivisions of the section, and the implicit true entries (the "+") for those conditions are a result of a series of blanket cross-references between those subdivisions. This type of Application Requirement is easily accounted for by the organizational system, if the classes follow the logical principles of classification. Inclusion of the Requirement as a datum places a burden on the organizational system, just as Cumulative Requirements do.

The following paragraphs illustrate a Synthetic Requirement (also see table 4.16):

> All frame components assumed not to be part of the seismic resisting system shall satisfy the minimum reinforcement requirements specified in Chapters 7, 10, and 11 of Ref. 11.1.

Table 4.15 Sample Decision Table for an Application Requirement Equivalent to a Simple Organizational Network

		1	2	3	4	E
1	Seismic performance category = A	Y	-	-	N	
2	Seismic performance category = B	-	Y	-	N	
3	Seismic performance category = C	-	-	Y	N	
4	(Seismic performance category = D)	-	-	-	+	
5	Category A design and detailing requirement = satisfied	Y	+	+	+	
6	Category B design and detailing requirement = satisfied	.	Y	+	+	
7	Category C design and detailing requirement = satisfied	.	.	Y	+	
8	Category D design and detailing requirement = satisfied	.	.	.	Y	
1	Structural design and detailing requirement = satisfied	X	X	X	X	
2	Structural design and detailing requirement = violated					X

Table 4.16 Sample Decision Table for a Synthetic Requirement

		1	2	3	4	5	E
1	Component = concrete frame component that is not part of SRS	N	Y	Y	Y	Y	
2	Reqt for minimum reinforcement of chap 7, 10, 11 of Ref 11.1 = satisfied	.	Y	Y	Y	Y	
3	Nonlinear behavior required to satisfy deform compatibility reqt = true	.	N	Y	N	Y	
4	Axial force due to all loads > 0.10 (FC)(AG)	.	N	N	Y	Y	
5	Special concrete flexural member lateral reinforcement reqt = satisfied	.	.	Y	.	.	
6	Special concrete beam column lateral reinforcement reqt = satisfied	.	.	.	.	Y	
7	Ordinary concrete beam column lateral reinforcement reqt = satisfied	.	.	.	Y	Y	
1	Category C and D non-seismic resisting concrete requirement = satisfied	X	X	X	X	X	
2	Category C and D non-seismic resisting concrete requirement = violated						Y

> If nonlinear behavior is required in such components to comply with Sec. 3.3.4(C), the critical portions shall be provided with Special Lateral Reinforcement as required for Special Moment Frames in Sec. 11.7.1(B) and/or Sec. 11.7.2(C) in these provisions.
>
> All frame components resisting axial compressive forces greater than 0.10 $f'_c A_g$ shall conform to the requirements of Sec. 11.6.2. [7]

The application of the Requirements referenced by conditions 5, 6, and 7 of the decision table is too complex for efficient use of the organizational system and is better handled as a decision table. But the more fundamental reason for including this as a datum is that it is effectively a new Basic Requirement. It names a THING that is different than that of the ingredient Requirements, and simply uses the ingredient Requirements as vehicles to deliver the appropriate REQUIRED QUALITY.

The distinction between Application Requirements and Synthetic Requirements occasionally may not be as clear as the other distinctions. Furthermore, this distinction is equivalent to separating the organizational system from the decision table system. It appears that the boundary between those situations best handled within the organizational system and those requiring a decision table depends primarily on whether the Requirement is really equivalent to a Basic Requirement (i.e., is a Synthetic Requirement) and secondarily on the context of the particular classification scheme. For example, table 4.15 obviously is handled better in the organization system, and table 4.16 obviously needs to be a decision table (even though the first condition could be treated in the organizational system).

Table 4.17 is an example of a Requirement that could be treated as either a Synthetic or an Application Requirement. It could be thought of as being equivalent to a new Basic Requirement, although the case for such treatment is not as clear as it was with table 4.16. The complexity of the logic reflected in the table indicates that it should exist as a decision table rather than being imposed on the organizational system. The table also illustrates a situation sometimes encountered in the representation of existing provisions; a complex Requirement that is not traceable to any single textual location, but is pulled together from several disjoint locations. It is never stated explicitly in the text of the provisions it represents, yet it is extremely important to the objective of judging compliance [50].

Mixed Requirements place a unique burden on the organizational system. Consider the following provisions (also see table 4.18):

> The following special requirements for concrete or composite concrete and steel piles are in addition to requirements for piles in the code administered by the Regulatory Agency.

Table 4.17 Sample Decision Table for an Application or Synthetic Requirement Inferred from Disjoint Provisions

		1	2	3	4	E
1	Quality assurance plan required = true	Y	Y	N	N	
2	Quality assurance plan acceptance requirement = satisfied	Y	Y	.	.	
3	Quality assurance plan compliance requirement = satisfied	Y	Y	.	.	
4	Mechanical/electrical equipment testing required = true	Y	N	Y	N	
5	Mechanical/electrical testing plan acceptance requirement = satisfied	Y	.	Y	.	
6	Mechanical/electrical test compliance requirement = satisfied	Y	.	Y	.	
	**					
1	Quality assurance requirement = satisfied	X	X	X	X	
2	Quality assurance requirement = violated					X

Table 4.18 Sample Decision Table for a Mixed Requirement

		1	2	3	4	5	6	7	E
1	Foundation structural components include concrete or composite concrete and steel piles	N	Y	Y	Y	Y	Y	Y	
2	Embeddment of pile reinforcement in pile cap ≥ Minimum development length	.	Y	Y	Y	Y	Y	Y	
3	Pile type = uncased concrete	.	Y	-	-	-	-	N	
4	Pile type = metal cased concrete	.	-	Y	+	-	-	N	
5	Pile type = filled steel pipe	.	-	N	Y	-	-	N	
6	Pile type = precast concrete	.	-	-	-	Y	-	N	
7	Pile type = precast prestressed concrete	.	-	-	-	-	Y	N	
8	Category B uncased concrete pile requirement = satisfied	.	Y	.	.	.	.	.	
9	Category B cased concrete pile requirement = satisfied	.	.	Y	.	.	.	.	
10	Category B steel pipe pile requirement = satisfied	.	.	.	Y	.	.	.	
11	Category B precast concrete pile requirement = satisfied	.	.	.	.	Y	.	.	
12	Category B prestressed concrete pile requirement = satisfied	.	.	.	.	.	Y	.	
	**								
1	Category B foundation pile requirement = satisfied	X	X	X	X	X	X	X	
2	Category B foundation pile requirement = violated								X

> The piles shall be connected to the pile cap by embedding the pile reinforcement in the pile cap for a distance equal to the development length as specified in Chapter 11. [7]

The bulk of the decision table representing this provision is an application Requirement of the simple type, with the Requirements referenced in conditions 8 through 12 corresponding to the " . . . following special requirements . . ." Condition 2 corresponds directly to the second of the quoted sentences, and it is a Basic Requirement in its own right. The burden placed on the organizational system is that this datum is impossible to class completely relevantly. The Application Requirement is general while the Basic Requirement is specific. The normal tendency is to class the datum by the general classifiers, but explicit access to the Basic Requirement is then lost, which is contrary to the objective of good organization. Mixed Requirements can involve Cumulative Requirements, Synthetic Requirements, and Application Requirements of both the simple and complex types.

Several observations are possible on the interface of the organizational system with the datum identification and derivation system based on the types of Requirements just described. First, the definition of datums is susceptible to implicit expression of one particular arrangement of a standard, which constrains the organization system from effectively developing optional arrangements. Second, it seems possible to avoid unintentional incorporation of a given arrangement in the datums by enforcing some restrictions *a priori* on the definition of datums. The pertinent restrictions are as follows:

1) Mixed Requirements should not be used.

2) Cumulative Requirements should not be used.

3) Multiple Requirements should be used only in those instances in which the constituent Basic Requirements would be most likely to remain together in all practicable arrangements, and then only with caution.

4) Application Requirements of the simple type should not be used.

These restrictions mean that datums representing Requirements would be representing Basic Requirements, Synthetic Requirements (which are equivalent to Basic Requirements), or Application Requirements that are too complex to depend upon the organizational system for clear representation. This is to promote access. The situation is somewhat analogous to the restriction that the actions of a decision table evaluate one datum only, which means that the decision table is uniquely identified with one node in the information network and "false" ingredients are avoided. The restrictions do not mean that datums or conditions of decision tables would never contain the same, or similar, information as the classification system. In situations that would not be efficiently handled by an organizational system alone, Requirements frequently make use of datums that may also serve as classifiers. As stated, the boundary defining such situations is not rigid. Furthermore, decision tables representing Determinations are also seen to contain such information. In general, decision tables would not include information that could be likened to the "selection" of a THING or REQUIRED QUALITY, but such information is sometimes necessary to sort between different rules for the evaluation of the same datum, be it Requirement or Determination.

4.1.4 Characteristics of Determinations

A study of the organization of standards for purposes of arrangement need not include Determinations, because strictly speaking, all the Determinations can be arranged solely from the information network once given the arrangement of all Requirements. (Because all terminal nodes on the information network are Requirements.) Determinations are one step further removed from the judging of compliance with the standard than the Requirements are; they are generated as a means for judging Requirements, which in turn are used for means for judging compliance. Nonetheless, it is important to classify Determinations because, in general, they should be indexed and in some instances they also are included in studies of arrangement.

In some "procedural" standards Determinations occupy the bulk of the standard. In fact, for some standards the Determinations are the most distinctive features of the standard, because many standards frequently share the same rather general Requirements for safety, strength, serviceability, and so on. Provisions such as the following (also quoted earlier)[7]:

> . . . total lateral seismic force determined in accordance with the following formula: $V = C_sW$ [7]

thus are very important. These important Determinations are frequently placed in conspicuous locations that do not correspond to any normal arrangement of the information network. In some instances the arrangement of a Determination is important because it is cross-referenced and used in a great many other provisions. Typical examples are provisions establishing the scope of a standard or establishing important categories of subjects. Thus, the organizational system must include Determinations for indexing and, frequently, also for outlining.

For several reasons, a study of the grammatical structure underlying Determinations does not seem warranted. It is important to recall that in many of the case studies in which a standard was represented with decision tables, a number of the provisions were ignored on the basis that they did not contribute to establishing the data structure for the purpose of judging compliance with the standard. For example, recall the portions of the provision dealing with "base shear" that were ignored or discounted in a section 4.1.1 ("Every building shall be designed to resist..."). This liberty taken with the original textual expression of such provisions is likely to make any thorough study of the underlying structure of the verbal expression a futile effort. In fact, the formal logic of the decision table appears to be a more appropriate structure for such provisions, and it does not seem to offer a basis for classification. The decision table representation of Determinations frequently tends to include fragments of widely scattered sentences, which further tends to complicate any grammatical analysis. For example, recall the following provision, the second portion of which is a Determination (also quoted in section 4.1.1):

> The area of a one-story building shall not exceed the limits set forth in Table 5-C except as provided in such 506, nor the limits specified in Chapter 16. [126]

The following provision found in a completely separate chapter of the standard also influences the same value and therefore is part of the same Determination:

> . . . the area may be increased by 50 percent when the maximum travel distance specified in Section 3302(d) is reduced by 50 percent. [126]

In fact, the second quoted provision is the basis for rules 8 and 9 in the decision table of table 4.2, even though the text provided no cross-reference.

One way of approaching a basic classification of Determinations is by looking at the characteristics of the value itself. One way of distinguishing among values is by whether they are numerical or not, another way is by whether they are continuous or discrete, and for discrete values by the number of possible values in the set. Thus, the base shear calculated from the formula referenced above is an example of a continuous numerical value, whereas the allowable area referenced above is a discrete numerical value (it is discrete because the table referenced in the quoted provision presents a discrete set of allowable areas). Some discrete values are not numerical, for example, the "Seismic Performance Category" defined in the new seismic design provisions [7], may have the values of A, B, C, or D. Some nonnumerical discrete values are of a two-value kind--either true or false (for example, the scope of the standard as applied to a particular object). In theory any discrete value can be treated as a Boolean vector in which the value must take one and only one of the discrete elements.

Although classification of Determinations by the symbolic type of value is useful from an analytical point of view, it does not appear useful for organizational purposes. Another approach is to consider that the values of all Determinations are qualities or measures of a quality, and then establish a taxonomy of qualities and measures. Such a taxonomy is most useful if it is relatively independent of the basic classes of THINGS and REQUIRED QUALITIES used for classifying Requirements. Classes that proved useful in the analysis [50] of the new seismic provisions are as follows:

1) requlatory parameters, such as scope, seismic performance category, etc.

2) basic physical measures, such as area, height, etc.

3) functional parameters, such as performance level, occupancy potential, etc.

4) structural response parameters, such as effective seismic forces, drifts, vibration characteristics, etc.

Additional attention is placed on such categories in the section 4.3.

There are other relevant ways to classify Determinations that can be of use in both indexing and arranging. For example, a Determination can be classed by:

1) The THING that the value is a quality of--for example, classing the provision for area by "building," or classing the provision for base shear by the "seismic resisting system," or the provision for the required level of inspection by the Process of "quality assurance" of which it is a characteristic.

2) The Process that the value would normally be derived in--for example, classing the base shear as "analysis,"

3) The REQUIRED QUALITY that the value is used for--for example, classing the base shear by "strength", or even "safety."

4) The Human that evaluates the value--for example, classing a provision for the determination of what is an essential facility by the term "building official" or the provision for base shear by the term "structural engineer."

In summary, determinations can be classified based on the characteristics of the qualities and measures that they establish. A Determination can also be classified by the fundamental categories used for the Requirements within its global dependence.

4.2 Principles for Classification

As stated in section 3.4, three issues related to coordinated principles for classification are raised in this study: 1) the overall structure of the classification, 2) the extent to which logical rigor is necessary, and 3) the detailed rules for classing provisions for both outlining and indexing.

The first issue is now easily treated. Faceted classification systems are the apparent choice over strictly logical systems because the organizational system has multiple purposes. Faceted systems similarly are the apparent choice over numerical techniques of classification because the number of relevant characteristics for a provision is small.

The remaining issues are treated by proposing and discussing two sets of principles: one for the development of a classification and one for the classing of provisions. The former is applicable to both the development of a new standard and the rearrangement of an existing one, while the latter applies only after a set of provisions exists.

The principles are intended to make possible the attainment of the objectives for an organization identified in section 1.2. Not all the objectives are addressed by these principles; some are only addressed by the techniques for outlining and indexing that are described in the following chapters. It is pertinent to recall that the objectives are stated in terms of headings while the present principles are in terms of single classifiers. In many instances headings are a combination of several classifiers. The techniques of outlining are designed to provide such combinations as will meet the objectives when working with a classification system that follows the principles stated here.

The principles account for the differences between the two functional types of provisions, Requirements and Determinations. They also account for the differing nature of the two primary organizational tools derived from the classification, outlines and indexes. The basic presentation is organized for outlines of Requirements, and the appropriate exceptions are noted for the other situations.

At this point it is appropriate to adopt as a guiding principle the following advice offered in a report on a classification system for the construction industry in England [1]:

> ... conventions should be flexible in order to cope with new ideas... (thus)... conventions should attempt to identify the fundamental concepts which underlie our knowledge.

4.2.1 Development of a Classification

The basic structure for a faceted classification is that of several independent fields, each composed of logically structured facets. The fields and facets will usually be drawn from a set of fundamental categories, such as those postulated for bulding design in section 4.3. There are five important principles governing the grouping:

1) There must be at least two independent fields: one for THING and one for REQUIRED QUALITY. Relevant classification of Requirements requires this as a minimum. There may be more than one independent field for each of these, such as fields for Physical Entity and Process for THING, or Performance Attribute and Limit State for REQUIRED QUALITY. There also may be independent fields for Determinations. The fact that the fields are assumed to be independent facilitates the construction of a classification and allows great freedom in constructing alternative arrangements.

2) Each facet must be a strictly logical tree. That is, the logical principles of classification, mutual exclusion and collective exhaustion, must be satisfied for each and every grouping within a facet. This is not typical of faceted systems, but at the most detailed level, logical rigor is necessary to satisfy two of the objectives of organization, Unique and Complete. Thus, for this application of faceted classification the distinction between facets

and fields provides a convenient and well defined point at which the logical principles can be applied.

3) A field may have any number of facets, and each facet, except the root facet, must be a logical subdivision of some other classifier in the field. In order to provide an outline that is Unique, Complete, and Graded, it must be possible to combine the facets in at least one way to produce a single logical tree for the entire field. Thus the potential connections between facets must be stated explicitly. A corollary of this principle along with the first principle suggests that the same facet not be used in more than one field, although duplicate facets could be used if the situation merited it.

4) The maximum number of siblings at any level should not exceed a reasonable estimate of the span of immediate memory of the user. Work in cognitive psychology suggests that "seven plus or minus two" is an appropriate number for maximum number of siblings from which a logical choice can be made [87, 91, 139].

5) The facets should promote an even division of the scope of the standard. Note that this tends to discourage the most rigorous logical classificatory scheme, bifurcation, in favor of coordinate series. The logical rigor necessary to deliver the objectives Unique and Complete take priority over this principle.

The third principle merits additional discussion. One use of the classification system is to build an outline, which must maintain the logical principles. Thus it is necessary to avoid the possibility of a tree containing siblings that are not mutually exclusive and to avoid any closed meshes. The first concern is the more substantive because in practice the second rarely arises. The first concern is satisfied by allowing only one facet to be appended to any one classifier.

Further study of this issue leads to an interesting fact. Consider the example taken from the analysis of the new seismic design provisions [50] shown in table 4.19. Two of the three facets apply to the classifier "Building Part," which is a terminal classifier on another facet (not shown). The third facet applies to the combination of the classifiers "Seismic resisting" and "Component." As is frequently the situation, it is necessary to be able to subdivide a single classifier (Building Part) by more than one facet. To accomplish this logically, the subdivisions are applied successively rather than concurrently. Thus the second facet is applied to the terminal classifiers of the first facet, as shown in the left column of table 4.19b. The right column shows a variation that maintains logical rigor but it does so by dividing the facet "Function of Building Part" into "subfacets." This subdivision of facets is quite useful in building outlines, and will be discussed further.

The example shows several other interesting features. Note that the name of a facet disappears when it is appended to another classifier. In this sense, the name of a facet is "transparent." Also note that

Table 4.19 Alternate Logical Combinations of Facets

a) three logical facets, shown in indented outline format

- Function of Building Part
  - Structural
    - Seismic resisting
    - Non-seismic resisting
    - Foundation
  - Nonstructural

- Scale of Building Part
  - System
  - Component
  - Material

- Type of Seismic Resisting Component
  - Frame
  - Shear panel

b) Two possible ways of combining the three facets into a tree

- (Function of Building Part)
  - Structural
    - Seismic resisting
      - System
      - Component
        - Frame
        - Shear panel
      - Material
    - Non-seismic resisting
      - System
      - Component
      - Material
    - Foundation
  - Nonstructural

- (Function of Building Part)
  - Structural
    - System
      - Seismic resisting
      - Non-seismic resisting
      - Foundation
    - Component
      - Seismic resisting
        - Frame
        - Shear panel
      - Non-seismic resisting
      - Foundation
    - Material
  - Nonstructural

facets need not be appended at all possible locations--"Scale of Building Part" is not appended to "Foundation" in the left column of figure 4.19b, and the second level subfacet of "Function of Building Part" is not appended to "Material" in the right column of table 4.19b. In addition, a single facet may require the combination (or absence) of several other classifiers to be relevant, as the third facet in table 4.19a. Techniques for combining facets are discussed in more detail in the description of outlining methods in subsequent chapters.

The division of a facet that is shown in table 4.19 illustrates that the fundamental unit of the classification is a single set of sibling classifiers connected to their parent classifier. This unit is the smallest unit that preserves the logical principles; hereafter it is termed a nuclear tree. For purposes of combining classifiers into an outline, a facet may be divided into its constituent nuclear trees. Thus the logical structure of a classification may be summarized:

1) A nuclear tree is the smallest logical unit.

2) A facet consists of one or more nuclear trees and is the largest logical unit. It may be subdivided into smaller logical units at any time.

3) A field consists of one or more facets and does not necessarily maintain logical rigor. Each field is considered to be an independent classification.

Useful classifications that violate the logical principles abound (see section 4.4 for examples). It is important that each deviation from the logical principles be soundly based and that the classification can still be interpreted to give an unambiguous indication of the applicability of a particular requirement to a particular class. The logical principles can be relaxed when classifying for the purpose of indexing, but it is more convenient to treat that in the principles for classing provisions.

In addition to the principles for grouping, care must be taken to provide relevant and progressive ordering of the classification. As stated in section 3.2.2, classes in a series may be ordered by a quality different from that used for class membership or by the degree to which they possess the quality for class membership. Attention is directed to the list of ordering principles discussed in that section and to the ordering principles of Vickery discussed in section 3.2.4.

4.2.2 Classing Provisions

There are five principles pertinent to the association of classifiers and individual provisions for the purpose of constructing outlines. Note in the following discussion that significant differences exist when classifying for the purpose of constructing an index.

1) Each requirement must be classed according to THING and to REQUIRED QUALITY.

2) No provision may be classed by more than one classifier from any one facet.

3) Each classifier associated with a provision must be the most detailed that includes the scope of the provision.

4) All terminal classifiers must be associated with at least one provision.

5) It is permissible to establish a priority among the classifiers associated with a provision.

The first principle assures relevance based on the model of the underlying structure of Requirements. Since no such model exists for Determinations, no comparable principle exists. It is frequently appropriate to make and follow an ad-hoc principle in the classing of Determinations in an attempt to be systematically relevant. In the analysis of the seimsic design provisions [50], each Determination was classed by a field for the "Type of Derived Measure" and by a field for "Processes" (the latter association based on the Process in which the value would normally be determined). Other ad-hoc principles for determinations are possibly more appropriate in other situations. Additional examples of relevant classes for Determinations are given in section 4.1.4.

The second and third principles are simply corollaries of the logical principles. It is frequently useful to violate both of them when classing for the purpose of indexing. Consider a provision that applies to both the "Seismic resisting" and "Foundation" parts of the "Structure", but not to the "Non-seismic resisting" parts (refer to the facet shown in table 4.19a). Outlining has the function of finding a single best location in a linear list for a provision. Thus, according to the logical principles, the provision must be classed as "Structure". Indexing has the function of getting a user to a provision from any relevant starting point, thus the provision is most appropriately classed "Seismic resisting" and "Foundation." It is fairly simple to account for different ways of classing the same provision and to call on the appropriate classification for a given purpose.

The fourth principle prevents useless detail in the classification system. The fifth principle is useful in some of the methods for outlining described in chapter 2. It does not appear to be particularly useful in indexing and was not considered for such use in this study. In the light of the basic structure of classification developed in this chapter, it appears that such a priority will be of the most use when applied to classifiers from different facets in the same field. Thus a provision classed as "Structure", "System", and "Seismic resisting" from the facets in Table 4.19a (with the first facet subdivided as previously described) might be more appropriately placed in one or the other outlines of Table 4.19b based on the priority given to "System" and "Seismic resisting".

4.3 Basic Categories

As stated in chapter 3, the establishment of a set of basic categories is important because it gives a firm starting point for the development of a classification for a particular standard. In this section a set of such categories appropriate for use in design standards for buildings are named and explained, with some discussion of their importance, application, and interrelationships. It is expected that the same or similar categories would be appropriate for standards that pertain to other subjects or that are broader in scope than building design. The objective in presenting these categories is to provide an aid for the development of a relevant and meaningful classification. They should be reviewed for any specific application. The indiscriminate application of these or any other basic categories is unwise.

The structure of a Requirement put forth in section 4.1.2 provides the basis for deriving basic categories. There must be at least two major independent fields, at least one for the THING (subject) and at least one for the REQUIRED QUALITY (predicate). Both of these are too general for the present purpose; their major constituents, which may be fields or facets are the real interest.

Before introducing any list of categories, a short discussion of definitions is appropriate. Recall that the definition of "thing" according to Webster includes "quality." The similar terms quality, property, characteristic, and attribute are distinguished as follows, according to Webster. Quality is the most general term, applying to that which is predicable of anything as one of its characteristics. Property is a peculiar or special quality, attribute is an essential, inherent, or necessary quality, and characteristic is particularly appropriate for classification as it is a quality or property distinguishing an individual, group, or type [131].

Because the classification of things is based philosophically on their qualities, care must be taken to distinguish the qualities used to classify the subject of a Basic Requirement and the qualities found in the predicate. The term Descriptive Quality is meant to imply the former while REQUIRED QUALITY is used for the latter. In an effort to remain consistent with previous work, the term Property is used only in the context of Descriptive Quality and the term Attribute is used only in the context of REQUIRED QUALITY.

Figure 4.4 summarizes the basic categories proposed for use with Requirements. The lower portion of the figure consists of categories that are useful in particular situations; thus they are termed facets. The figure is not a logical classification itself, and there may well be other usable facets not included. A brief discussion of each category with selected examples follows.

4.3.1 Categories of THINGS

Evidence for three principal categories of THINGS is given in section 4.1.2. They are: Physical Entities, Human Entities, and Processes.

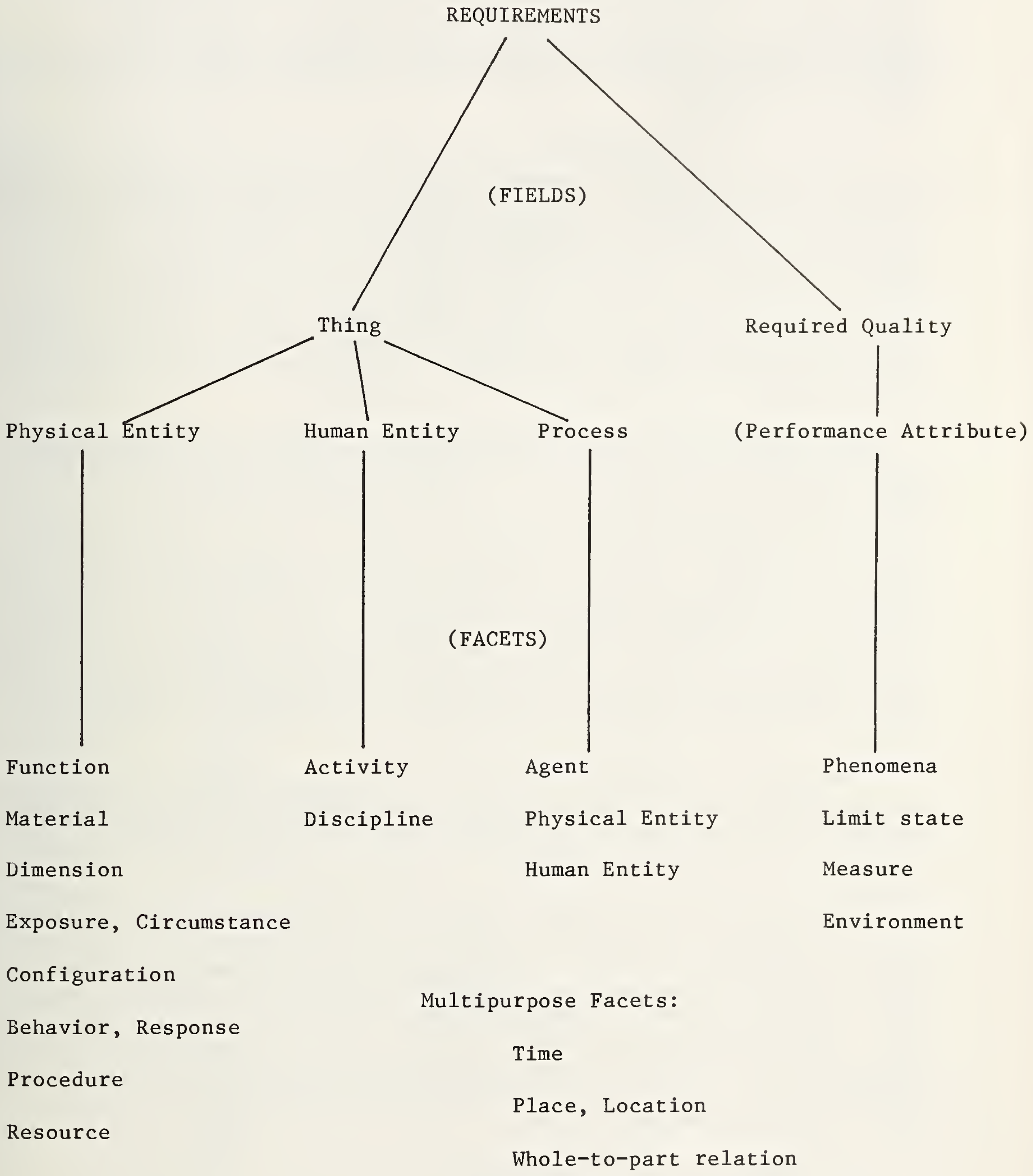


Figure 4.4 Basic Categories (Fields and Facets) for Requirements

(By definition, Human Entities are distinct from all other Physical Entities.) As discussed there, some potential for ambiguity exists between Human Entity and Process for some provisions and between Physical Entity and Process for other provisions. These ambiguities are usually resolvable by considering the most relevant REQUIRED QUALITY. There are, however, additional important issues to be illuminated concerning the proposed three categories of THINGS.

First is the philosophical issue that no event or Process is independent of a Physical Entity. Indeed, events are known by the characteristics of the objects involved. Note that "substance" is the first and most important of Aristotle's categories and that Kant derived his categories from the judgement of sensations of objects [62] (see section 3.2.3). This would indicate that Process is simply a way of identifying a Physical Entity and would be a facet, not an independent field. In fact, Process is proposed as a category with the tacit assumption that all processes involve a Physical Entity (for example "construction" in a building standard must imply "construction of the building").

The other issue arises from performance theory. In the context of buildings, user needs are related directly to the physical behavior of the built elements (see figure 4.5). In most situations, the only level of control necessary and sufficient to provide for the user needs would be on the built element (Physical Entity). (Occasionally, controls are necessary on the user.) However, building standards contain Requirements that do exert controls on Processes and Human Entities for many reasons that are beyond pure performance theory. For example, the omission of reinforcement in concrete, an error of Process, is unlikely to be detected until disastrous physical behavior occurs, thus a control on the Process is prudent. Likewise, some standards address issues beyond user needs, for example the safety of the participants in the Process.

Therefore, the inclusion of more than Physical Entity as the subject of Requirements appears well grounded. The three categories are useful, the most important factor. They are distinct enough for selection of the most relevant class in a particular situation.

In addition to the examples cited in section 4.1.2, other classifications of information relating to buildings offer evidence for these categories of THINGS. In review of the organization of several documents, Kapsch points out that building regulations typically contain provisions addressing what is built (Physical Entity) and how it is built (Process) [64]. Table 4.20 shows the basic categories for information about buildings and other constructed works proposed in a study performed by the British government [1]. An earlier European system for buildings only, the "SfB", has three categories [83]:

1) Element (parts by function)

2) Constructions (parts by how they are made)

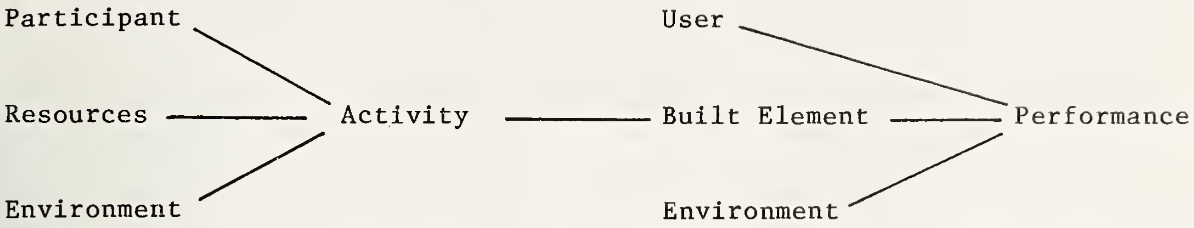


Figure 4.5 Idealization of Information Areas for Performance Theory

Table 4.20 Classification Framework for the British Construction Industry (Reference 1)

A (THINGS)

Construction Works
Parts of Works
Equipment
Constitutents
Agents
Users
Environment

B (ABSTRACTS)

Properties
Processes
Operations and Activities

C (PERVASIVE ABSTRACTS)

Space
Time

D Peripheral Subjects

3) Resources (including materials, etc.)

A British adaptation of this, the "CI/SfB" adds two new categories and combines two, as follows [27]:

1) Built Environment

2) Building Elements

3) Construction Forms and Materials

4) Activities and Requirements

Classification of information has been important enough to the building community that it was one of reasons behind the founding of the International Council for Building Research Studies and Documentation (CIB) [66]. Reference 72 gives historical insight on the problem.

A most important factor in the classification of THINGS, shown as a "multi-purpose" facet in figure 4.4, is the whole-to-part relationship. A common characteristic of Physical Entity classifications is the large number of levels that are typically related in the fashion of "system-subsystem-component-constituent." Similarly, Processes can be divided into stages (references 122 and 125 give examples for structural engineering). Great detail in the whole-to-part naming of Physical Entities is most common in prescriptive standards; it is somewhat antithetical to the concept of a performance standard. The proper mix of whole-to-part classification and classification by the more implementation independent facets is critical to the successful use of a standard. Where innovation is desired, too much whole-to-part naming can restrict the freedom. Where easy judgement of compliance is desired, a classification strongly tied to the most common implementations is desirable. As the facets of figure 4.4 are reviewed, the whole-to-part relationship should be kept in mind.

There is considerable richness in the way the Physical Entities are named and classified, as evidenced by the large number of Descriptive Qualities in common use. Following are examples to help define those facets in figure 4.4:

- Function: the division of stairs into required exits and supplemental exits; the division of a structural system into seismic resisting and non-seismic resisting subsystems.

- Material: the division of structural components into wood, steel, reinforced concrete, masonry, etc.

- Dimensions: the division of buildings into one-story and multi-story.

- Exposure (Circumstance): the division of components into those in contact with corrosive fluids, those in contact with soil, etc.

- Configuration: the division of stairs into straight, spiral, etc.

- Behavior: the division of materials into brittle and ductile.

- Procedure: the division of concrete components into those cast-in-place and those precast.

- Resource: the division of energy by coal, oil, nuclear, solar, etc.

- Time: the division of buildings into existing and proposed.

- Place: the division of components by location on the top story, below grade, etc.

Examples of the facets for human entity follow:

- Activity: the division into users of buildings and agents of the building process; the division of users into occupants, maintenance crews, neighbors, etc.

- Discipline: the division of designers into architects, structural engineers, mechanical engineers, etc.

- Time: the division of licensed designers by the date of registration.

- Place: the division of licensed professional designers by state of registration.

Similar examples for facets of process are:

- Agent: the division into natural processes, such as corrosion, and human operations, such as welding.

- Physical Entity: the division of pile driving by types of piles, for example; open steel section, closed steel section, timber, and precast concrete piles.

- Human Entity: the division of quality assurance activities into those carried out by the designer, the regulator, the contractor, etc.

- Time: the division of structural design into conceptual design, analysis, proportioning, detailing, etc.

- Place: the division of welding into shop welding and field welding, or downhand and overhead welding, etc.

Some useful facets may be a combination of some of the above. For example, in standards for the structural design of buildings a class of "stress states" is very useful (axial stress, flexural stress, shear stress, etc.) for grouping components. The class can be thought of as Function, Circumstance, or Behavior, or some combination of them. The fact that it doesn't fit clearly into figure 4.4 does not deter its usefulness. The study by the British government [1] noted three general types of "operational" categories in addition to their "fundamental" categories shown in table 4.20:

1) use of a different characteristic to cut across other categories
2) use of a different grouping within the same category for a new purpose
3) combination of two or more categories.

The new seismic provisions for buildings [7] provide an example of a unique derived class that is very relevant, meaningful, and useful in the organization of the document. Much use is made of the "Seismic Performance Category," which is defined function of the seismicity of the site of the building (Exposure) and the use of the building (Function).

There is a great deal of interrelation among the facets for the three categories of THINGS. It should be noted that the ambiguity between the Human Entity and Process categories discussed in section 4.1.2 is almost entirely confined to those Human Entities that can be described as an agent of the building process and those Processes that can be described as human operations. The classification by Human Entity becomes particularly useful when the requirement is establishing responsibility.

4.3.2 Categories of REQUIRED QUALITIES

The classification of REQUIRED QUALITIES does not show the richness of the classification of THINGS, but this is not an indication of simplicity. A strong case has been made by many [38, 14] that all Requirements should be related to a Performance Attribute, but as discussed in section 4.1.2, many Requirements can not be related confidently to a single Performance Attribute. Thus the first problem with the categories shown for REQUIRED QUALITY in figure 4.4 is that the principal category, performance attribute, is not usable for some standards. Another factor that contributes to the relative difficulty of developing a classification of REQUIRED QUALITIES is that the whole-to-part relation is rarely applicable, making the logical structure entirely dependent on the characteristics of the REQUIRED QUALITIES.

In spite of the problems, the classification of REQUIRED QUALITIES is worthwhile for two reasons. It is necessary to allow full and relevant classification of Requirements, without which the access function of the organizational system would be severely hampered. It also allows a completely independent arrangement of a standard that concentrates on the objective rather than the subject, which is quite desirable for some individuals and some uses. Thus the categories of figure 4.4 are presented with the knowledge that difficulties will be encountered in the classification of REQUIRED QUALITIES. No better general purpose categories have been found, however. A supplementary basis for classifying qualities, both REQUIRED and Descriptive has proved to be of use on some recent case studies. It is described subsequently.

Performance theory allows the establishment of a hierarchy of Requirements, or of REQUIRED QUALITIES, that extends from the needs of a user of a building to the characteristics of a building. The hierarchy of provisions described in chapter one (performance requirement--performance criterion--evaluation--commentary, see figure 1.2) is directly comparable to a hierarchy of REQUIRED QUALITIES such as performance attribute--phenomenon--measure. The term Performance Attribute is the most general REQUIRED QUALITY. It is almost always closely related to a user need or activity, and it is rarely directly measurable. Very typical examples of Performance Attributes for buildings include "safety," "health," "durability," "functionality," "economy," "privacy," "security," "accessibility for the handicapped," etc. Logical classification of concepts like these is not easily accomplished, and as already pointed out, logical classification of prescriptive provisions by such concepts is frequently impossible because a single provision frequently is related to more than one of the concepts.

Much as Performance Attributes focus user needs for the purpose of building design without restricting design freedom, Phenomena can be used to increase the specificity over that offered by Performance Attributes without resorting to prescriptive provisions. "Fire," "windstorm," "earthquake," or "temporary storage overload" are Phenomena that might relate to the Performance Attribute "safety." Likewise "heavy traffic," "solar degradation," and "insect infestation" might relate to "durability," and "forced entry" might relate to "security." Some Phenomena can be conveniently subdivided for further distinction among Requirements: for example, dividing "fire" into "heat," "smoke," and "flame." In most of these examples, the REQUIRED QUALITY is not the occurrence of the Phenomena, but rather the prevention or control of the Phenomena. Phenomena are frequently the most relevant way of classifying REQUIRED QUALITIES in performance oriented Requirements and criteria (recall that a criterion is defined as a measurable Requirement).

A very similar way of stepping down the performance hierarchy is through the use of Limit States as classifiers of performance criteria. In the context of structural engineering, a Limit State has been defined as an event that may cause the loss of a Performance Attribute, either by its occurrence or by its amplitude [53, similar definitions in 5, 6, 133]. The Limit State concept may be worthy of wider application, because it appears to be quite relevant for performance standards and also quite meaningful. Examples of Limit States and their related Performance Attributes are "collapse of a building" (safety), "vibration of a floor" (comfort), and "cracking of a water tank" (function).

It appears that Limit States are generally more scheme dependent than many of the Phenomena cited in the prior paragraph, and furthermore that they apply well only to those Requirements with Physical Entities as subjects. In several studies [50 is an example] Limit States have been related explicitly to performance Requirements, not simply to Performance Attributes, thus showing the event positively linked to a THING, usually a Physical Entity.

The advantage of conceptualizing a performance standard through the use of Limit States is that the event usually is measurable in some way, thus leading to unambiguous evaluation of the standard. In the context of structural engineering again, Limit States frequently lead to a design equation involving an action upon a structure and a quality of the structure, for example the stress caused by a load and the strength available to resist that stress. Classing such design equations according to the Limit State is frequently more useful than classing them according to the quality directly required of the Physical Entity, such as strength. It is necessary to avoid confusing a Phenomenon with a Property when using Phenomena and Limit States to class Requirements, however. For example the Phenomenon "vibration" must be regarded as distinct from the Property "susceptibility to vibration."

Measure is shown in figure 4.4 as a reminder that not all requirements can be classified through the performance hierarchy, and that even for those that can there exists a more basic facet for the REQUIRED QUALITY. In this sense, Measure is all encompassing from the most fundamental qualities, such as existence through the more remote or accidental qualities like "Circumstances." When performance concepts fail to group REQUIRED QUALITIES adequately, more arbitrary distinctions are frequently useful. Figure 4.6 shows examples of two bifurcations that have proven useful in several studies: "physical" versus "non-physical" (herein called "social") qualities, and "measurable" versus "abstract" qualities. As shown in figure 4.7, the same structure can be useful for Descriptive Qualities.

Environment, Time, and Place are frequently useful facets for REQUIRED QUALITIES in a fashion analogous to the concept of Property facets for THINGS. They can be used to define the situations in which certain qualities are pertinent or required. For example, the load sources on a structure (the environment) are useful to define the situations in which specific Limit States are to be prevented. Similarly Time is used to group Requirements concerning the phenomenon vibration into those for "transient" vibrations and those for "steady state" vibrations.

4.3.3 Extension to Determinations

The basic categories are useful for classifying Determinations in addition to requirements, although the basic reasoning presented for the categories is strictly applicable only to Requirements. Inclusion of the Determinations among the provisions classified has a greater effect on the logic of associating classifiers and provisions than on the categories used in the classification. Determinations relate well to Physical Entities, Processes, and Measurable Qualities, but no underlying structure is identified for Determinations that leads to the relations between classes shown in figure 4.4. Obviously, the value established by the determination is,

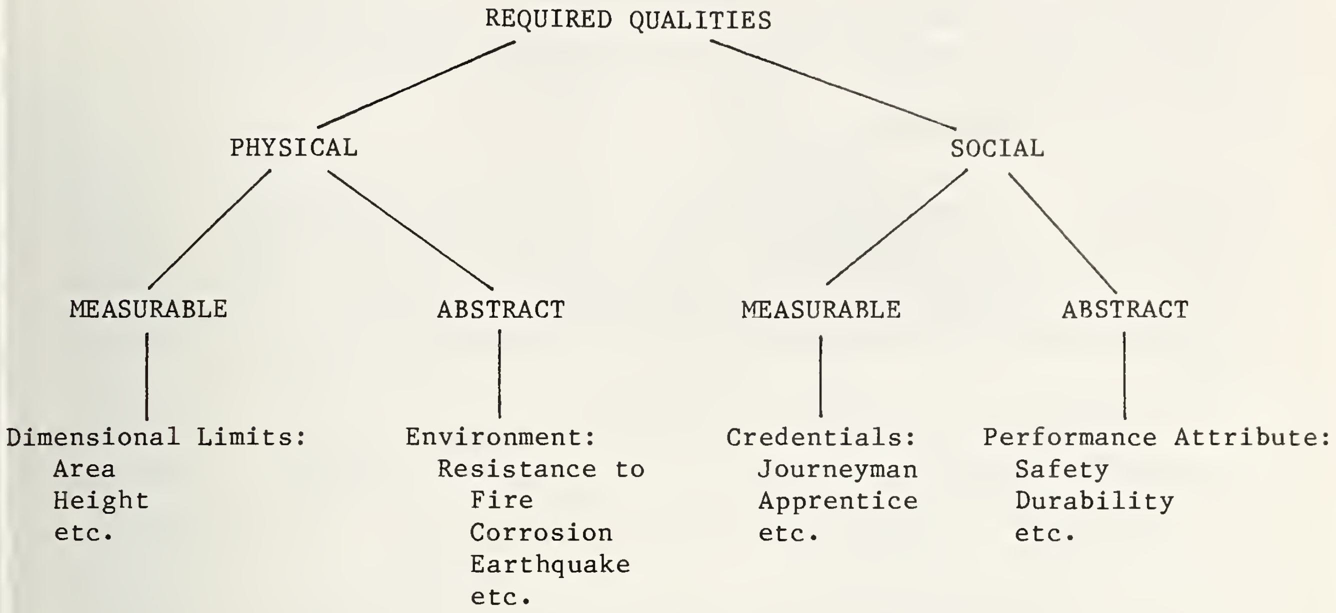


Figure 4.6 Useful Secondary Classification of Required Qualities with Examples

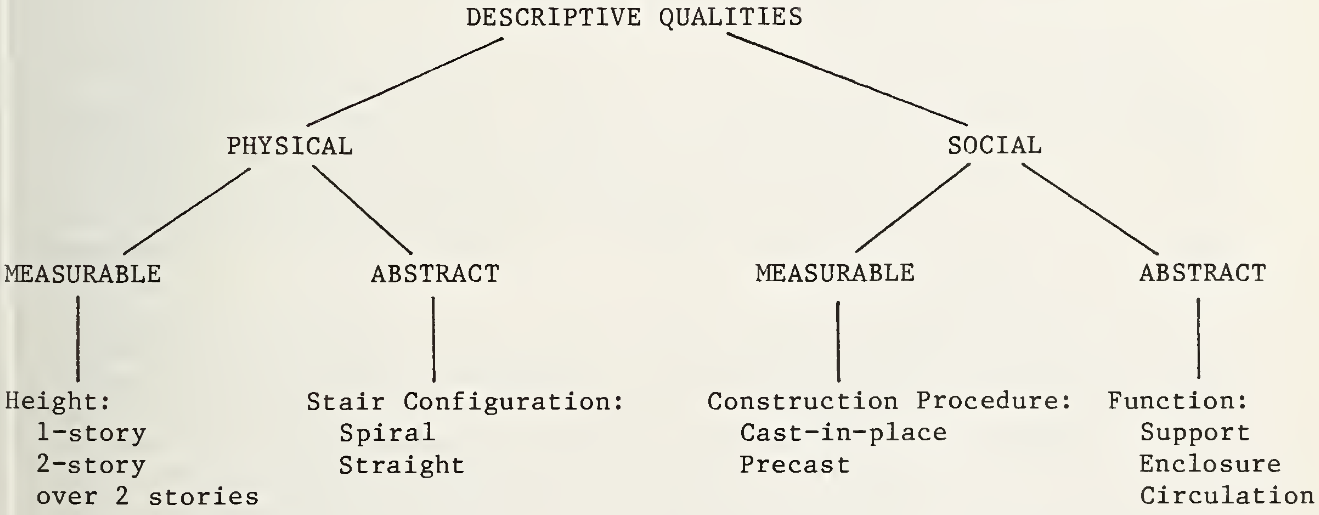


Figure 4.7 Useful Secondary Classification of Descriptive Qualities with Examples

by definition, a measurable quality. As stated in section 4.1.4, an independent field for such qualities is frequently useful.

4.4 Example Development of a Classification

Table 4.21 shows a classification that was developed as this study progressed. It does not meet all the criteria and concepts put forth in this report, in part because it was completed and published well before this study became complete and in part because a general treatment such as this one usually needs tailoring for specific applications. (Table 4.21 is a very slightly modified version of the classification published in reference 50.)

The classification was used for 242 Requirements and 163 Determinations It has five fields, as follows:

1) A Physical Entity field that is used to class the subject of all Requirements.

2) A Process field that is used to class all provisions; for some Requirements it represents the subject, and for other provisions it represents the stage in which the provision would normally apply.

3) A Required Quality field that is used to class the predicate of all Requirements.

4) A Limit State field that is used to class the predicate of some design oriented Requirements; it is a supplementary field.

5) A Derived Value field that is used to class all Determinations.

The Physical Entity field contains 21 facets; each of the other fields contains one facet. The example illustrates well the richness of a Physical Entity classification as compared to a REQUIRED QUALITY classification. No field is offered for Performance Attribute, because nearly all of the Requirements relate to "Occupant Safety." A small number of the Requirements pertain to "Community Safety."

The classification contains some classes that merit specific comment, particularly with regard to the logical principles. A violation of the logical principles that frequently is useful is the incomplete class. Consider the facet for "Concrete Pile Construction." Obviously, the division of precast piles is incomplete, because "non-prestressed" is not used, but it must exist. The reason that such a classifier was not used is simple: it was never needed. No Requirement in the provisions being analyzed ever applied to nonprestressed precast piles without also applying to prestressed precast piles, while several Requirements did apply to prestressed precast piles but not to nonprestressed precast piles. This particular type of incompleteness is common in the analysis of an existing set of provisions.

Table 4.21 Classification for Seismic Provisions for Buildings (Reference 50)

a) Physical Entity Field (21 facets -- * denotes root of a facet)

- *Building
  - Whole Building
  - Part of Building
- *Seismic Performance
  - Category A
  - Category B
  - Category C
  - Category D
- *Seismic Hazard Exposure
  - Group III
  - Groups I and II (Not Used)
- *Existence of Building
  - Proposed (New)
  - Existing
- *Material Nature of Bldg Part
  - Material Generic
  - Material Specific
- *Scale of Building Part
  - System
  - Component
  - Material
- *Function of Building Part
  - Structural
    - Seismic Resisting
    - Non-Seismic Resisting
    - Foundation
  - Non-Structural
    - Architectural
    - Mechanical/Electrical
- *Structural Components
  - Connection
  - Member (Not Used)
- *Materials of Construction
  - Wood
  - Steel
  - Reinforced Concrete
  - Masonry
- *Type of Seismic Resisting Comp
  - Frame
    - Moment Frame (Unbraced)
      - Ordinary Moment Frame
      - Special Moment Frame
    - Braced Frame
  - Shear Panel
    - Shear Wall
    - Diaphragm
- *Frame Components
  - Beam
  - Column
  - Joint
- *Part of Shear Panel
  - Boundary Member
  - Web (not used)
- *Part of Foundation
  - Soil
  - Foundation Structure
    - Pile
    - Non-Pile (not used)
- *Non-Structural Components
  - Equipment
  - Anchorage
- *Wood Design Method
  - Conventional
  - Engineered
- *Part of Wood Shear Panel
  - Framing (Wood)
  - Sheathing
    - Plywood
    - Diagonal Board
    - Other Sheathing Material
- *Reinf Concrete Constituents
  - Concrete
  - Reinforcement (Concrete)
    - Lateral Reinforcement
    - Longitudinal Reinforcement
- *Concrete Pile Construction
  - Cast-in-Place
    - Cased
    - Uncased
  - Precast
    - Prestressed
    - Non-prestressed (Not Used)
- *Masonry Constituents
  - Masonry Unit, Mortar, Grout
  - Reinforcement (Masonry)
- *Masonry Construction
  - Unreinforced
  - Stacked Bond
  - Hollow Unit Masonry
- *Type of Member Stress
  - Axial Stress
  - Flexural Stress
  - Shear Stress
  - Torsion Stress

Table 4.21 Continued

b) Process Field

- Building Processes
  - Regulation
  - Design
    - Site/Soil Investigation
    - Conceptual Design
    - Analysis
      - Seismic Load Analysis
        - Equivalent Lateral Force
        - Modal
        - Soil-Structure Interaction
      - Member Force Analysis
    - Detailed Design
  - Construction
  - Quality Assurance
    - Planning (QA)
    - Inspection
    - Testing
  - Use
    - Alteration
    - Repair
    - Change of Use
    - Hazard Abatement
      - Qualitative Evaluation
      - Analytical Evaluation

c) Required Quality Field

- Required Qualities
  - Physical Qualities
    - Measurable Physical Qualities
      - Existence of Objects
      - Reference Standards
      - Details
        - Quantities and Dimensions
        - Configuration (Arrangement)
      - Strength Required
      - Stiffness/Flexibility Reqd
    - Abstract
      - Integrity
      - Interrelationship
  - Social Qualities
    - Existence of Process
    - Method
      - Techniques
      - Principles and Assumptions
    - Documentation

d) Limit State Field

- Limit States
  - Collapse
    - General Failure
    - Progressive Failure
    - Overturning
  - Hazardous Damage
    - Collision
    - Drift, Excessive
    - Access/Egress Blocked
    - Component Failure
    - Component Anchorage Failure
    - Secondary Hazard
  - Ground Rupture
  - Dysfunction of Designated Seismic System

e) Field for the Type of Determination

- Derived Values
  - Basic Physical Measures
    - Height
    - Length
    - Weight
    - Time
  - Regulatory Parameters
    - Scope
    - Ground Motion
    - Classification of Objects
  - Functional Measures
    - Performance Level
    - Occupancy Potential
    - Capacity
    - Soil Properties
  - Structural Response
    - Response Modification
    - Damping
    - Period of Vibration
    - Seismic Base Shear
    - Seismic Story Force
    - Seismic Force Effect
    - Seismic Deflection
    - Seismic Drift
    - Combined Force Effect
    - Second Order Effects
    - Non-structural Seismic Force

It is not wise to allow it in the initial formulation of a set of new provisions until the scope has been completely defined. Other classes with only one active classifier occur for similar reasons ("Seismic Hazard Exposure," "Part of Shear Panel," etc.).

A considerable amount of similar condensation from a purely logical structure has occurred in other classes, for example, the constituents and types of masonry construction. As a matter of fact, the classifiers for types of masonry construction ("Unreinforced," "Stacked Bond," "Hollow Unit Masonry") are not mutually exclusive. This drawback was accepted in this study because the infrequent use of those classifiers did not justify the amount of hierarchical structure required to maintain the logical principles. Other classes such as the facet for "Type of Member Stress" exhibit a potential ambiguity in that any given member could be subjected to more than one of the types of stresss listed. The provisions [7] never refer to a member under combined stress situations, therefore, in the context of these provisions once again, the logical principles are intact.

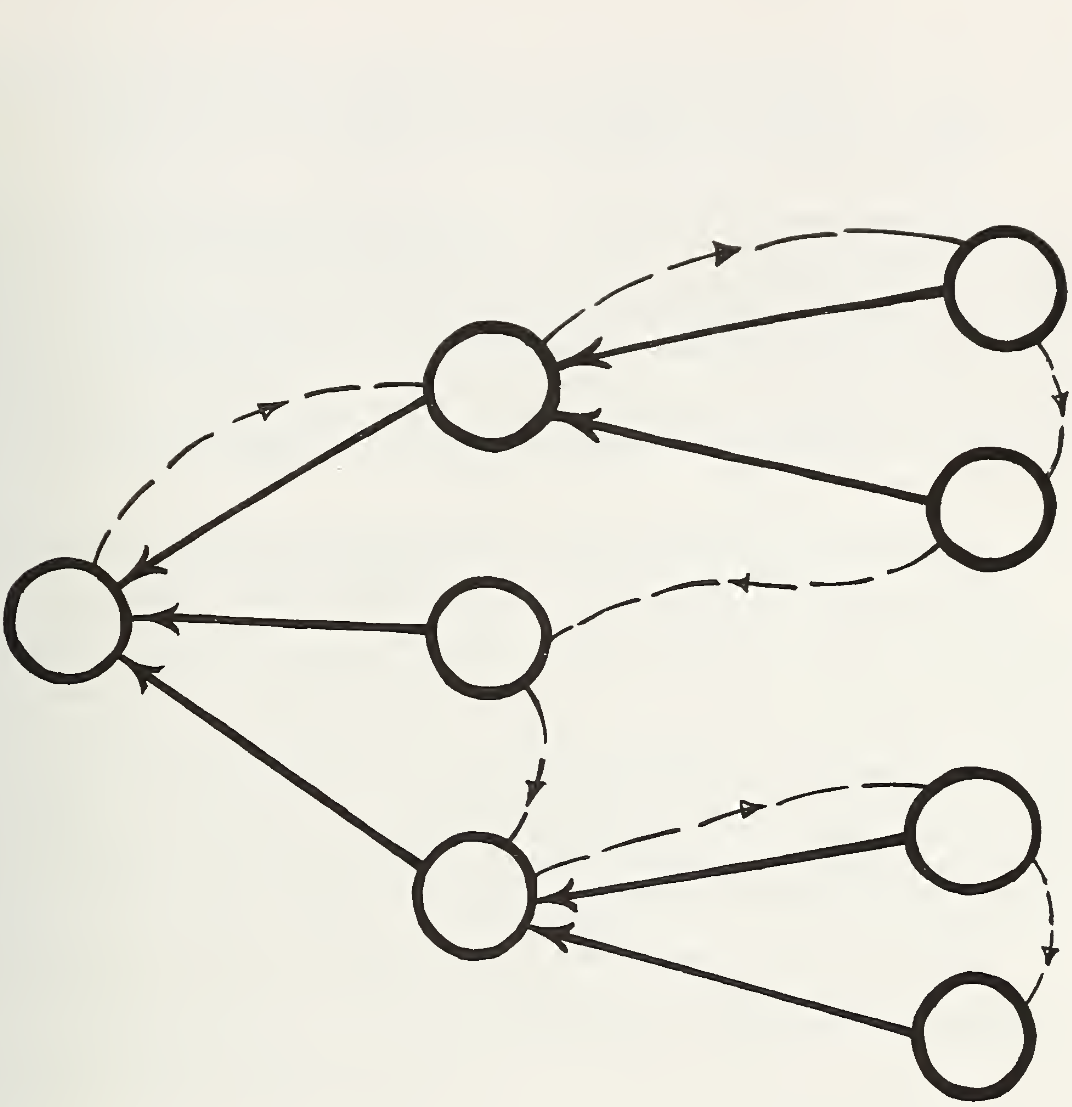
Some classes exhibit an unusual structure or relation between the classifiers at a given level. For example, the classifiers of the class "Scale of Building Part" ("System," "Component" and "Material") are related to each other in that a component may be made of a material, a system may contain several components which are in turn made of materials, etc. The use of this facet follows the policy that a provision is classed according to which REQUIRED QUALITIES are specified. Thus, "Components" or "Materials" may be specified as REQUIRED QUALITIES of a "System," however, "Components" would not be specified as REQUIRED QUALITY of a "Component."

Each of these deviations from the logical principles of classification were made to reduce the cumbersome nature of the classification system and make it more streamlined and useful. They are examples for which the objectives Relevant, Meaningful, Intelligible, Minimal, and Even are more important than Unique, Complete, and Graded. For balance, an example of the importance of the logical principles is appropriate.

A simple example from the new seismic design provisions [7] shows one of the problems for readers caused by the violation of the logical principles. In those provisions, all buildings are classified according to "Seismic Performance Category" (the derived class mentioned in section 4.3.1). In addition, many, but not all, of the Requirements are explicitly identified as pertaining to one of the "Seismic Performance Categories." (There are four: A, B, C, and D.) Thus the following situation exists: four sets of buildings (categories A, B, C, and D) and five sets of Requirements (those identified for each category plus those not identified for any category). Furthermore, a relationship exists between the categories such that a building must not only satisfy the Requirements explicitly identified for its category, but also all of those for the "lower" categories (A is the lowest, D is the highest).

The problem arises in determining just what the difference is between those Requirements that are not explicitly associated with any category, which might conceivably apply to all buildings, and those for "Category A," which do apply to all buildings because of the cascading relationship. One potential solution is to assume that there are no differences, that both sets apply to all buildings, and that the inconsistency is only apparent, not of substance. Such apparent problems occur often in the analysis of standards, but in this instance the problem is real. Only a few of the Requirements that are not explicitly associated with any category are intended to be applied to "Category A" buildings. The bulk of them are intended to apply to "Category B" (and "higher") buildings, but this is not stated explicitly in the provisions. Complete classification of the Requirements according to the logically sound "Seismic Performance Category" would have prevented the ambiguity and confusion created by attempting to apply five sets of Requirements to four sets of buildings.

Facing page: "Conditional order" provides a top-down sequence; "formulation" is a top-down procedure.



Chapter 5

FORMULATION OF A STANDARD

The model for the organization of a standard makes possible a technique for the formulation (initial development) of a standard that gives some assurances of relevance and completeness. The technique is a rather formal approach to the development of a standard that can provide much needed direction to the early deliberations in the process. It can, for example, prevent the drafting of detailed technical Determinations before agreement is reached on the Requirements that furnish the need for such Determinations. The technique cannot possibly incorporate all the considerations necessary in the proper development of a standard, however, and it should not be used in a heavy handed manner that stifles creativity. Seiss's 1960 comment [114] on standard's writing committees is an appropriate caveat:

It is neither necessary nor desirable that all members of such bodies approach the task of writing a code from the same point of view or with the same philosophy.

The technique includes two processes: the classification of the scope of the standard, including both subject (THING) and objective (REQUIRED QUALITY), and the identification of basic Requirements through the construction of an outline from the classification. The technique relies upon performance concepts and is related to methods used in the past to formulate performance standards. Also, the technique is related closely to the outlining methods described in chapter 6, the fundamental difference being that this technique is designed for identifying Requirements rather than rearranging provisions.

5.1 The Use of Performance Theory in the Organization of a Standard

As stated, any Requirement, whether performance or prescriptive, is in some way related to a Performance Attribute, but the relation may not be apparent for an existing prescriptive Requirement. For a new standard this relation must be clear for all types of Requirements, for it is the basis for the creation of new Requirement. Performance theory deals with this relation between Performance Attributes (or the reason for a provision) and Requirements, thus it is of fundamental importance in establishing the scope of a new standard. This makes performance theory useful in organizing any type of standard. Performance based techniques used for organizing standards are therefore of interest, and are examined here.

This is not advocacy of writing all new standards as performance standards. These are advantages and disadvantages for either approach. The advantages of performance standards are:

- The flexibilty allowed encourages innovation and efficiency in production.

- They are generally not so technical as to preclude understanding by nontechnical people.

- They are inherently more "complete" in that they may be applied in uncommon situations.

- They are less likely to contain irrelevant Requirements.

- They don't require continual updating to keep pace with changing technology [40] (at least in theory, reference 120 indicates that anything formally codified is subject to change).

- They may, in some instances, be brief and clear.

On the other hand, the advantages of prescriptive standards are:

- Judgement of compliance is straightforward.

- Less expense is incurred in translating the Requirements into an acceptable solution.

- There is less discretion allowed, consequently less potential for acceptance of a solution that is contrary to the goals of the standard.

- It is possible to require the use of the most modern and best possible specific solutions.

- They may, in some instances, be brief and direct, in part because some Performance Attributes may be implicitly assumed through the use of conventional products and processes [140].

Thus there are good reasons for formulating standards that are not oriented to performance.

It should be recalled that normally no standard is purely performance or prescriptive, but that a continuum exists between the extremes. The challenge is to discover the proper use of performance theory in the formulation of any kind of standard. Kapsch [63] has described the process used at the National Bureau of Standards in the development performance standards for housing [103], for solar heating and cooling systems [59], and for public office buildings [55] as follows:

1) The building systems are defined (i.e., the Physical entities are classified).

2) The user needs are defined (i.e., the Performance Attributes).

3) A two dimensional matrix is constructed with the two classifications as the indices.

4) Performance requirements are defined at the appropriate intersections.

5) One or more performance criteria are developed for each performance requirement.

6) Evaluation procedures and commentary are prepared for each performance criterion.

7) The matrix is used to arrange and index the standards.

Note that this process makes explicit use of the RCEC format described in chapter 1. The idea that a complete performance statement must contain a performance requirement, a performance criterion, and an evaluation procedure specifically remedies the primary disadvantage of performance standards, the difficult evaluation of compliance.

Kapsch criticized the process as suffering several problems [63], the foremost of which can be described as problems in attaining a logical classification with only two linear lists available to capture the classification. An additional problem that becomes apparent upon reading performance standards produced through this process and arranged in a strict RCEC format based on the matrix is that many performance criteria and some performance requirements seem redundant. The matrix enforces a uniform scale of grouping of attribute and entity for all performance requirements, which may not be appropriate for each situation. In addition, the RCEC format does not recognize the possibility of a single performance criterion being applicable as a measure of more than one performance requirement.

It is also apparent that the process does not give any guidance for the development of a performance criterion, it only accounts for its place in the development. The hierarchy of Performance Attribute, Phenomenon or Limit State, and Measure included in the basic categories described in section 4.3.2 and taken from the work of Fenves, Rankin, and Tejuja [38] does give a rationale for the identification of performance criteria. This rationale can be worked into an extension of the process.

Not all performance standards are produced or organized in this fashion. A Presidential task force charged with review and reform of the safety standards of the Occupational Safety and Health Administration (OSHA) chose a performance approach for increasing safety in the workplace and alleviating problems associated with the existing prescriptive standards [54]. As an example they produced a draft of one part of the safety regulations that deal with the guarding of machines [32]. Their format is as follows [77]:

1) General Performance and Hazard Obligations

   - the definition of entities covered and attributes desired (a statement of scope and of performance requirements)

   - a comprehensive list of recognized hazards (a list of phenomena)

   - sufficient and insufficient general methods for safeguarding the hazards (a list of measures related to the hazards, and thus to the performance attributes)

   - special (prescriptive) rules for extreme hazards for which no discretion is allowed (a set of measures for specific physical entities with a tacit relation to performance attributes)

2) Guidelines for general approaches to the problem (a primer for information only, essentially a commentary)

3) Guidelines for regulatory compliance (a catalog of specific solutions deemed to satisfy the performance requirements)

One interesting point to note about this format is that it contains no criteria for quantitative evaluation of acceptable performance (or risk, in this situation), yet it does define Performance Attributes, Phenomena, and Measures in part 1. Another is the inclusion of mandatory prescriptive requirements alongside the performance statements in part 1. The manual of acceptable solutions contained in part 3 is something frequently advocated as a corollary to a good performance standard, but rarely is one actually produced.

Another example of a novel performance standard is one for the design and construction of foundations being prepared by the American Society of Civil Engineers. This standard has a small number of performance requirements and a much larger number of performance criteria, some of which are very general and others of which apply to only certain physical entities. Thus the following three-part format has evolved:

1) Performance requirements (almost a "purpose" statement)

2) General performance criteria

3) Criteria for specific situations

   - performance criteria

   - prescriptive criteria that may be used in lieu of checking the performance criteria

The formulation of this standard will be discussed in more detail in section 5.3 as an example of the application of the technique described in the next section.

5.2 Technique for Formulation

Two processes are involved in the technique: the construction of a classification and the construction of an outline. The overall technique consists of two phases, and iteration is normal. The first phase is the identification of Performance Requirements, as follows:

1) A "top-down" classification is constructed for performance Requirements. This obviously must include at least one field for THING and one for REQUIRED QUALITY. For a performance Requirement the REQUIRED QUALITY is a Performance Attribute.

2) An outline is constructed by appending the trees of classifiers into one large tree (hereafter called an organizational tree).

> The procedures for appending trees are described more fully in the following; they are similar to the procedures for outlining described in section 2.4. A performance Requirement is associated with each terminal node on the organizational tree.

This phase is obviously very similar to the "matrix" process described in the previous section. A significant difference is that more than two fields may be used. Multiple fields are possible for THING, for example one for Physical Entity and one for Process. Additional fields or facets are possible for the REQUIRED QUALITY also, but Performance Attribute must be the primary field. Other differences from the "matrix" are introduced through the use of trees rather than single level lists to represent the classification and the use of the organizational tree rather than the matrix to represent the combinations of classifiers.

The second phase is the identification of specific Requirements (the performance criteria in the context of a performance standard). The two processes are repeated:

1) The classification is extended, once again in a "top-down" fashion, to include classifiers relevant to performance criteria. Thus the REQUIRED QUALITY fields must be extended or supplemented with Phenomena and/or Limit States capable of providing unambiguous measures of the performance Requirements. It frequently requires that the classification for THING become more specific, and that context relations be established between Phenomena and THINGS.

2) A new organizational tree is constructed with the extended classification (it may only be an extension of the organizational tree for performance Requirements) and a specific Requirement is associated with each terminal node.

This phase is an extension beyond the "matrix" process described in the preceding section. It offers a rationale for the identification of performance criteria and other specific Requirements. Depending on the manner of extending the classification and constructing the network, it can produce organizations within or outside the RCEC format.

Top down classification is essentially a way of making decisions about scope, and then retaining those decisions as the criteria for completeness and relevance in the organization of a standard. It is practicable because the classification of the subjects and objectives of a standard gives automatically a classification for the THINGS and REQUIRED QUALITIES for the Requirements of the standard. The performance approach to the classification (first Performance Attribute, then Phenomena or Limit State) is simply the most relevant way of moving from the general and qualitative to the specific and measurable.

The purpose of constructing the organizational tree is to produce the combinations of THING and REQUIRED QUALITY classifiers that correspond to Requirements. The ordering of Requirements is not of particular concern at this stage, although the tool for reordering provisions is just an embellished version of the same organizational tree (see section 6.2), and thus some attention can easily be given to likely arrangements. The construction of the organizational tree is presented as a formal algorithm below and illustrated by examples in the next section. It is capable of being programmed as a computer aid. Such a computer aid has not been produced, however, because no instance of the algorithm's use has been of a scale warranting the effort.

The first step is to explode the trees of the classification into nuclear trees (recall that a nuclear tree is defined in section 4.2.1 as a classifier and its direct children). Thus, the process of appending trees into an organizational tree deals with one set of logical brothers at a time. The relations among and within facets are not forgotten in this step; each nuclear tree belongs to a certain facet and field, and the context conditions for a facet apply to each of the nuclear trees taken from it.

The second step is to select the root of one of the major THING or REQUIRED QUALITY fields as the root for the organizational tree. The process of appending begins with consideration of its first child, which is the first terminal node to be examined in step three.

The third step is the heart of the algorithm. Considering the classifiers on the branch from the root to any particular terminal node as the "stack", the stack and the remaining nuclear trees are examined to determine whether it is necessary, appropriate, or possible to append an additional nuclear tree. Because the model of a basic Requirement has a THING as subject and a REQUIRED QUALITY as predicate, it is necessary to append another tree if the stack does not contain the root nuclear trees for the field of THING and the field of REQUIRED QUALITY. It is often appropriate that the stack contains more than one classifier for either or both of these two categories. For performance criteria, for example, the REQUIRED QUALITIES should include both a Performance Attribute and a Phenomenon or Limit State. The possibility of appending a nuclear tree is governed by the following rules:

1) No classifier may be repeated on any branch.

2) The hierarchy of the original classification must be preserved. Thus no classifier may be appended on to a branch "below" (that is, farther from the root) any classifier at a lower level in its original facet. Note that this does not prevent the separation of a parent and child by many levels of classifiers from unrelated facets.

3) The context rules for a facet apply for every nuclear tree taken from the facet. Note that the root facet of a field has no context rules.

Rules 1 and 2 deal with the objective Graded and rule 3 deals with Relevant. Unique and Complete are assured, to the extent that the classification is logical, by appending only one nuclear tree at a time.

If the conditions are met, the nuclear tree is appended by "burying" the parent at the current terminal node and then proceeding to its first child. The third step is then repeated. If more than one nuclear tree are possible candidates, the most appropriate should be selected from due consideration of the consequences of the subsequent execution of step three. If no possibility of appending another tree exists, proceed to step four.

The fourth step is to select the next appropriate action once a branch is terminated. If the terminal classifier has a sibling remaining, proceed to it and execute step three for the new branch. If no sibling remains, examine the parent in the organizational tree (note that, in general, this would not be the parent of the nuclear tree). If that classifier is not a root of a field, the fourth step is repeated. If that classifier is a root and if no other fields remain that can be used to start a new tree (or a new "trunk"), then the algorithm is completed. If such fields remain, the root can be appended and the algorithm continued from step two.

Figure 5.1, table 5.1, and table 5.2 summarize the construction of the organizational tree. If a Requirement is associated with each branch in the tree, that is with each terminal node, then the Requirements cover the scope of the tree. If the tree contains each field in the classification such that each terminal classifier from the root nuclear tree of each THING field is combined with each terminal classifier from the root nuclear tree of each REQUIRED QUALITY field, the tree covers the scope of the classification. Note that different THING fields need not be combined and that descriptive QUALITY facets do not control any check for completeness. The order in which classifiers occur on a branch is not a factor in this technique (except for hierarchical considerations) although it is a consideration in some techniques for arrangement, as discussed in the next chapter.

5.3 Example Applications of the Technique

Two examples are presented to illustrate the technique described. The first follows the RCEC format for performance standards, and all the subjects are Physical Entities. The second departs from the classic model in several respects: the format is not RCEC, the subjects include building Processes, and procedural and prescriptive Requirements are included.

5.3.1 Innovative Residential Structures

This example is taken from a set of provisions prepared as a model performance standard for innovative for residential buildings [102]. The example is limited to those provisions within the domain of structural engineering.

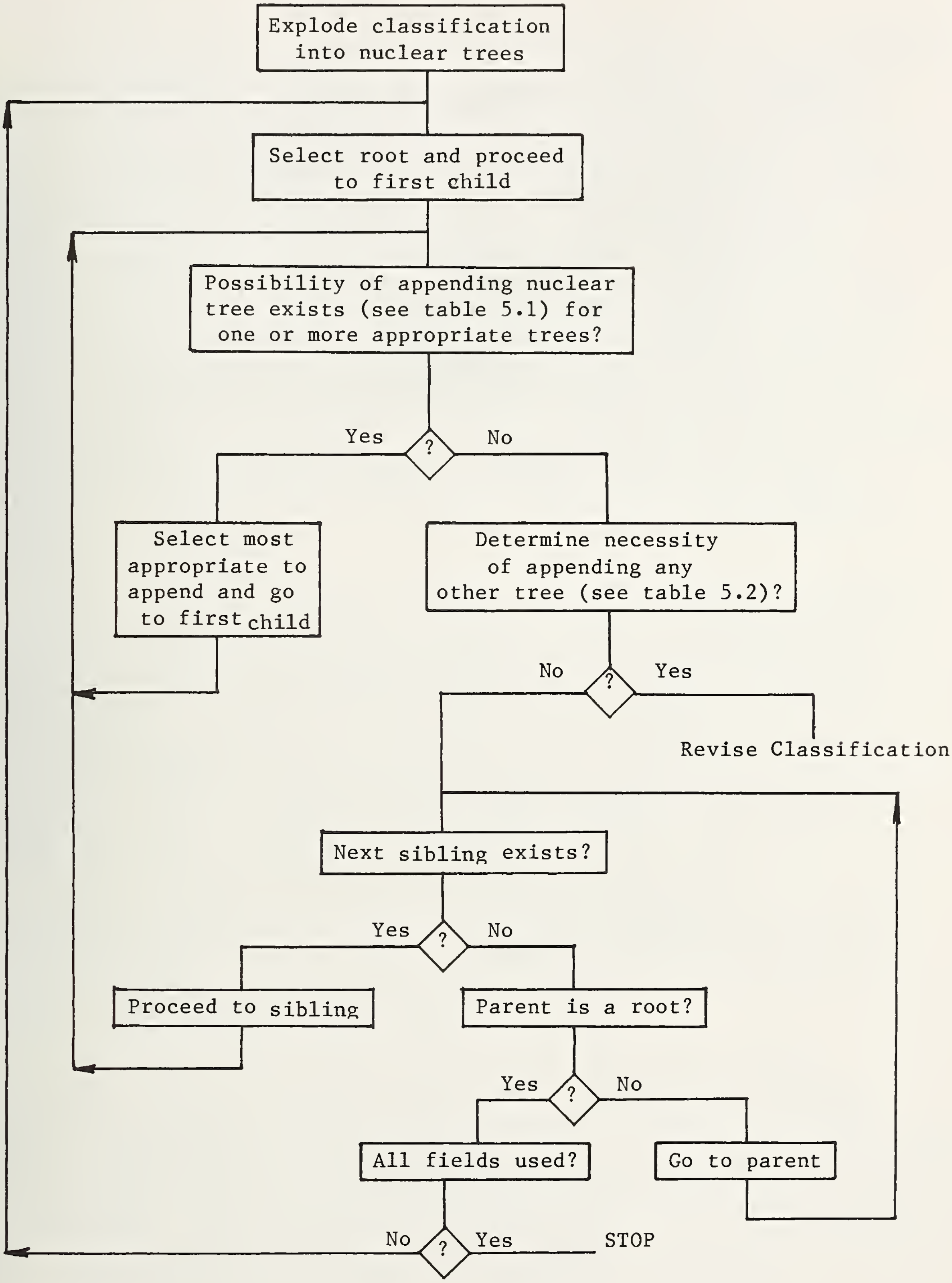

Explode classification into nuclear trees
Select root and proceed to first child
Possibility of appending nuclear tree exists (see table 5.1) for one or more appropriate trees?
Yes
?
No
Select most appropriate to append and go to first child
Determine necessity of appending any other tree (see table 5.2)?
No
?
Yes
Revise Classification
Next sibling exists?
Yes
?
No
Proceed to sibling
Parent is a root?
Yes
?
No
All fields used?
Go to parent
No
?
Yes
STOP

Figure 5.1 Construction of Organizational Tree for Formulation

Table 5.1 Check for Possibility of Appending a Nuclear Tree

		1	2	3	4
1	Classifier in the nuclear tree is in the stack	Y	N	N	N
2	Nuclear tree from same facet as some classifier in the stack <u>and</u> the nuclear tree is at same or higher level of that facet	.	Y	N	N
3	Facet context rules satisfied	.	.	N	Y
1	Can append				X
2	Can not append	X	X	X	

Table 5.2 Check for Necessity of Appending a Nuclear Tree

		1	E
1	Stack contains terminal classifier from root facet of a THING field	Y	
2	Stack contains terminal classifier from root facet of a REQUIRED QUALITY field	Y	
1	Must append		X
2	Not necessary to append	X	

The first step is to establish those classifiers necessary to organize the set of performance Requirements. Table 5.3 shows these classifiers in outline formt. The Physical Entities illustrate several important points. First, the scope of the entire set of provisions is limited to the structure of residential buildings. "Building" (standing for "residential building") is used as the most general Physical Entity. There are two subdivisions of buildings that are to be considered: "Structural System" is obvious, but "Interior Surfaces" is not. The latter is included because the provisions will address the structural aspects of interior surfaces. Note that there are many parallel subdivisions of a building that are not included ("heating system," "lighting," etc.) which do have structural features, but that "Structural System" and "Interior Surfaces" are defined as a complete expression of the scope of this set of set of provisions and thus establish the criterion for completeness that the organization must meet. Similarly, "Floors" and "Walls" are not the only possible interior surfaces of a building; others, such as "ceilings" are consciously excluded at this stage because "Floors" and "Walls" are defined as a complete set. Note that the terms satisfy the necessary principle of uniqueness, that is, "Walls" and "Floors" are distinct and unlikely to be confused.

The Performance Attributes, "Safety" and "Serviceability," illustrate the richness in meaning that some classifiers possess. In the context of design regulations for building structures, "Safety" is generally taken to mean life safety for occupants and neighbors of buildings. "Serviceability," however, means more to a wider range of people. For occupants and neighbors, it means that the behavior of the structure should not impair the functionality of the building or cause discomfort to the occupants. For owners, it means that the structure should be maintainable and durable. Thus, although "Safety" and "Serviceability" are meaningful in the sense that the intended audience for the provisions understand them, they are not necessarily simple words.

Two "Environments" are listed to show that arbitrary divisions are sometimes useful as long as they still satisfy the requisite properties for classifiers. Note that this division between "Force Loads" and "Other Agents" was not originally perceived but was entered in one of the several iterations necessary to conduct the study.

Four nuclear trees exist in the classification, with parents "Building," "Interior Surface," "Performance Attribute," and "Environment." One field exists for THING and one field exists for REQUIRED QUALITY, "Environment" being an auxiliary facet in that field. Table 5.4 shows an organizational tree in outline format generated from the two root nuclear trees. Four performance Requirements are possible, one for each terminal node. Table 5.5 shows an organizational tree incorporating all possible branches for the four nuclear trees and a column denoting performance requirements for discussion in the following paragraphs.

Table 5.3 Classification for Performance Requirements for Residential Structures

```
Building                              Performance Attribute
  Structural System                     Safety
  Interior Surface                      Serviceability
    Wall                              Environment
    Floor                               Force Loads
                                        Other Agents
```

Table 5.4 Simple Organizational Tree for Performance Requirements (Residential Structures)

```
Building
  Structural System
    Safety - - - - - - - - - - - - R
    Serviceability - - - - - - - - R
  Interior Surface
    Safety - - - - - - - - - - - - R
    Serviceability - - - - - - - - R
```

Table 5.5 Organizational Tree for Performance Requirements (Residential Structures)

```
Building
    Structural System
        Safety
            Force Loads  - - - - - - - R1
            Other Agents - - - - - - - X1
        Serviceability - - - - - - - - R2
            Force Loads /*
            Other Agents /
    Interior Surfaces
        Safety - - - - - - - - - - - - X2
            Force Loads /
                Floor /
                Wall /
            Other Agents /
                Floor /
                Wall /
        Serviceability
            Force Loads  - - - - - - - R3
                Floor /
                Wall /
            Other Agents - - - - - - - X3
                Floor /
                Wall /
```

* A "/" indicates that the scope of the requirement above will include this classifier, that is, the branch will not be subdivided beyond the preceding node without a "/".

The first Requirement is associated with the three classifiers "Structural System," "Safety," and "Force Loads." Using the model structure of a Requirement discussed earlier, the performance Requirement could be expressed as: "The structural system shall safely support all loads expected during its service life." The next entry, X1, shows that a Requirement could have been written for "the safety of structural systems under the action of other agents of the environment," but was not. On the surface this is incomplete. However, the effect of "Other Agents," such as "Heat," "Moisture," etc., on "Safety" is not direct when compared to "Force Loads" and in fact is coupled to the presence of "Force Loads." Because the effect of "Other Agents" on "Safety" is of a different order, and because it is coupled to "Force Loads," this effect is covered in the criteria related to Requirement R1. Decisions about the organization, such as this one on completeness, are not necessarily easy or quick. Complex physical behavior or arbitrary limits on scope frequently require extended deliberation and compromise. The outlining technique calls attention to potential missing provisions, such as this, and requires explicit decisions by the standards writers.

Requirement R2 is written without considering the distinction between "Force Loads" and "Other Agents." It is not necessary to use all facets when writing Requirements in many instances, and this is one. It will be shown that the two classes of environment are useful to group the criteria for R2 into two progressive sets, so their presence in the outline will be justified later. X2 shows that no Requirement is associated with the classifiers "Safety" and "Interior Surfaces." This is because "Interior Surfaces" (a wall surface does not include the entire wall) are not considered to present any hazard to life in the structural sense of their behavior. Other kinds of hazards, such as toxicity, are possible, but are outside the scope of these provisions because only Limit States of a structural nature will be included. Once again, the outline has identified a potential missing provision and caused the writer to explicitly consider the impact of leaving it out.

Given that the only Phenomena concerning "Interior Surfaces" to be considered by these provisions are those of a structural nature, one is realistically limited to considering "Force Loads." Other problems of "Serviceability," such as paint adherence, are outside the scope, thus, X3 is shown.

The longer organizational tree has the effect of refining the scope of the two Requirements dealing with "Safety" to exclude "Environments" other than "Force Loads." Otherwise the very simple outline of Table 5.4 is adequate. No need has been shown for the classifiers "Wall" and "Floor" in distinguishing among performance Requirements. They are used to distinguish among performance criteria, however.

Table 5.6 shows the additional classifiers necessary to organize the criteria. No new fields are added, only more specific facets, and each facet or classifier applies in a specific context, as shown. The first and most important facet is the Limit States. The RCEC format implies

Table 5.6 Additional Classifiers for Performance Criteria for Residential Structures

CLASSIFIER	CONTEXT
LIMIT STATE	
Failure	Rl, R3
Deflection	R2 + Force Loads
Drift	R2 + Force Loads
Vibration	R2 + Force Loads
Dimensional Change	R2 + Other Agents
Loss of Material	R2 + Other Agents
Material Change	R2 + Other Agents
TIME DESCRIPTORS FOR LIMIT STATES	
Load Occurrence	Failure
Expected Maximum	
Repeated	
Exceptional	
Vibration Duration	Vibration
Transient	
Steady State	
Deflection Duration	Deflection + Ductile
Short Term	
Long Term	
RESPONSE MEASURES FOR VIBRATIONS	
Transient Vibration Measure	Transient (Vibration)
Amplitude	
Damping	
Steady State Vibration Measure	Steady State (Vibration)
Acceleration	
Resonance	
ENVIRONMENT DESCRIPTORS FOR DRIFT	
Drift Loads	Drift
Wind	
Earthquake	
PHYSICAL ENTITY REFINEMENT	
Structural Parts	Dimensional Change
Member	
Joint	
STRUCTURAL RESPONSE DESCRIPTORS	
Deflection Compatibility	Deflection
Ductile	
Brittle	

that a Requirement is the appropriate context for a Limit State. Beside each Limit State is shown the Requirement(s) that it applies to, and in the case of those Limit States applying to Requirement R2, the particular "Environment" that is relevant. Thus, the Limit State "Failure" is associated with both Requirements R1 and R3, and the Limit State "Vibration" is associated with requirement R2 when "Force Loads" are considered.

The remaining facets relate to qualities and entities used to further define the Limit States, the Measures of the Limit States, and the THINGS. Each subclass is developed for association with some Limit State as shown. Thus, the class "Drift Loads" is developed for the Limit State "Drift," the class "Load Occurrence" is developed for the Limit State "Failure," the class "Deflection Duration" is developed for the classifier "Flexible," which is part of a class of THINGS that is developed for the Limit State "Deflection," etc.

In general, these additional classes are necessary to define the scope in the detail necessary to write precise criteria. Frequently the same distinction can be obtained in alternate ways. For instance, it is necessary to separate grossly different modes of failure of structural systems because the performance measures used in the criteria are different. In this example, the separation was accomplished by considering the different levels of probability that a given load would occur: 1) those expected to occur once in the life of the structure ("Expected Maximum"), 2) those expected to occur many times in the life of the structure ("Repeated"), and 3) those expected to occur in the life of only a very few structures ("Exceptional"). The order of classifiers was related to likelihood of being a controlling design criterion, rather than to a progressive order of the likelihood of load occurrence. It also would be possible to separate the criteria by having a class of failure types: 1) conventional failure, 2) fatigue failures, and 3) exceptional failures, such as progressive collapse. The two classes for separating the criteria would result in the same criteria being written. The former class was selected because it was felt that it was more relevant yet still meaningful.

Table 5.7 shows the organizational tree of classifiers and the corresponding outline of Requirements (performance Requirements and performance criteria). The network was generated in the same fashion as the one for performance Requirements. The outline is simply a condensation of the organizational tree with each heading corresponding to a Requirement. Note that in some instances "sub-criteria" are necessary to maintain the proper hierarchy, thus the format is not strictly RCEC. The headings enclosed in parentheses indicate that the subject matter is covered concisely in one requirement corresponding to the preceding heading. They are shown only to facilitate comparison with the organizational tree.

5.3.2 Foundation Standard

This example is presented to demonstrate application of the technique to a standard that does not follow the RCEC format. The example is more hypothetical than the previous example because the technique has been

Table 5.7 Organization Including Performance Criteria (Residential Structures)

ORGANIZATIONAL TREE	PROVISION OUTLINE
Building	
Structure	
Safety	
Force Loads	R1 Structural Safety
Failure	
Expected Maximum	C1.1 Resistance to Max. Load
Repeated	C1.2 Resistance to Rep. Load
Exceptional	C1.3 Resistance to Excep. Load
Serviceability	R2 Structural Serviceability
Force Loads	
Deflection	C2.1 Deflections Under Load
Ductile	
Short Term	(Short Term)
Long Term	(Long Term)
Brittle	(Brittle Materials)
Drift	C2.2 Lateral Drift
Wind	(Wind)
Earthquake	(Earthquake)
Vibration	C.2.3 Vibration
Transient	
Amplitude	2.3.1 Trans. Vib. Amplitude
Damping	2.3.2 Trans. Vib. Damping
Steady State	2.3.3 Steady State Vib.
Acceleration	(Acceleration)
Resonance	(Resonance)
Other Agents	C2.4 Service Environment
Dimensional Changes	
Members	2.4.1 Dim. Changes in memb.
Joints	2.4.2 Joints
Loss of Material	2.4.3 Corrosion
Material Changes	2.4.4 Material Changes
Interior Surfaces	
Serviceability	R3 Serviceability of Floors and Walls
Force Load	
Failure	
Floor	C3.1 Floors
Wall	C3.2 Walls

applied only to the initial formulation of the standard, which is still under development. The example thus lacks the refinement produced by iteration. Because the scope of the standard is large and there are many classifiers, the organizational tree is large; only an abbreviated presentation of it is made.

The chairman of the committee developing the standard has planned a three level approach to the criteria, as follows:

1) scheme-independent performance criteria

2) scheme-dependent design and procedural criteria clearly related to performance concepts

3) scheme-dependent prescriptive criteria

Satisfaction of the criteria at one level is to be deemed to satisfy the criteria at the higher levels. Thus the third level is in effect a catalog of simple solutions for the more abstract performance criteria. ("Criterion" is used in lieu of "Requirement" to emphasize that all the Requirements in the Standard are to be objectively measurable.) This approach shares several features with the performance approach proposed for use in OSHA standards discussed in section 5.1.

The THING classification for the standard is shown in table 5.8. The obvious difference between this example and the previous one is the inclusion of two independent fields of THINGS. The total range of THINGS is also much broader. The two fields are not really wholly independent, although they are assumed to be so for the present purpose. It is worthwhile to keep in mind that many or most of the criteria with a Physical Entity as THING will govern the Process of "design," and that all of the Processes are concerned with some Physical Entity. The three auxiliary facets in the Physical Entity field ("Soil Material Type," "Structural Material," and "Exposure") are very tentative.

The REQUIRED QUALTIY classification is shown in table 5.9. It is less complete and generally more tentative than the THING classification, and will be discussed in more detail. The Performance Attributes, which may be complete, are defined somewhat by the remarks in the table. Each of them is generally scheme-independent, and the writing of different performance Requirements for different THINGS would be repetitious. Thus it was decided at an early stage to abandon the RCEC format by writing just one performance Requirement for each Attribute and placing them together at the beginning of the standard as a preamble, or statement of purpose.

The Limit States shown in table 5.9 are undoubtedly incomplete. Some extension in terms of subdividing the ones shown will be discussed as the example progresses. Additions in breadth are also probable. For example, none of those given would pertain to adverse impact on the visual environment due to accumulation or dispersion of trash and other debris. The relations shown between the Limit States and the Performance Attributes

Table 5.8 Initial THING Classification for a Foundation Standard

Field 1 - Physical Entities

- Soil System
    - Unsupported Excavation
    - Level Fill
    - Sloping Fill
    - etc.
- Structure Supported by Soil
    - Shallow Foundation
        - Isolated Spread Footing
        - Combined Footing
        - Mat
    - Deep Foundation
        - Pile
        - Pier
    - Concrete Dam
    - etc.
- Structure Supporting Soil
    - Retaining Wall
    - Anchored Bulkhead
    - Braced Excavation
    - Buried Structure
    - Cofferdam
    - Tunnel
    - etc.

- Soil Material Type
    - Soil
        - Clay
        - Silt
        - Sand
        - Gravel
        - Mixed
    - Rock

- Structural Material
    - Concrete
    - Steel
    - Timber

- Exposure
    - Air
    - Fresh Water
    - Salt Water
    - Soil
    - Freezing

Field 2 - Processes

- Soil Investigation
    - Field Exploration
    - Field Testing
    - Laboratory Testing
    - Monitoring
- Construction Operations
    - Movement of Water
        - Drainage
        - Dewatering
    - Movement of Soil
        - Excavation
            - Open Cut
            - Drilling
            - Caisson Sinking
        - Backfill
        - Compacting Fill
        - Pile Driving
        - Blasting
        - Underpinning
        - etc.
    - Improvement of Soil
        - Compaction
        - Stabilization
        - Injection Grouting
        - etc.
    - Storage and Disposal
        - Water
        - Soil

Table 5.9 Initial REQUIRED QUALITY Classification for a Foundation Standard

Performance Attributes, with remarks

Safety - from death or personal injury.

Function - the ability of the foundation to perform its intended function without hampering any important adjacent functions.

Damage - the protection of property adjacent to or supported by the foundation from damage due to the behavior of the foundation.

Environment - the preservation of desirable environmental qualities from undesirable changes as a consequence of the foundation or its construction.

Limit States and Related Performance Attribute

	S	F	D	E
in solid media:				
Instability	X	X	X	
Deflection		X	X	
Vibration		X	X	X
in liquid media:				
Seepage		X	X	X
Flooding	X		X	X
Erosion			X	X
Sedimentation			X	X
in atmospheric media:				
Weathering	X	X	X	
Noise				X
Dust				X

Loads

Soil
Dead
Live
Hydrostatic
Hydrodyanmic
Adjacent Operations
Adjacent Foundations
Restraint
Earthquake
Wind
Snow

Geometric Measures

Vertical Cut Height
Sideslope Ratio
etc.

Mechanical Measures

Strength
Stiffness
Permeability
etc.

indicate another reason for abandoning the RCEC format. For those Limit States related to multiple Attributes, it is likely that the same criterion would have to be repeated. This is not universally true, because different levels of performance can be associated with different Attributes. It is common to specify different levels of reliability against "Instability" for the different Attributes "Safety" and "Function."

No relations between the Limit States and the THINGS are shown. Many exist, and will be detected in the initial construction of the organizational tree. For example, "Vibration" is a significant Limit State for the "Operations" of "Pile Driving" or "Blasting," but no criterion is needed for vibration when the THING is a "Retaining Wall." As another example, the Limit States are frequently too general for some THINGS and subdivision is necessary. The Limit State "Instability" might be divided into "Overturning," "Sliding," and "Structural Integrity" for the THING "Retaining Wall," and into "Landslide," "Bottom Heave," and "Liquefaction" for the THING "Excavating,"

Some Limit States are relatively independent of the THING, for example "Noise" and "Dust," in part because the Phenomenon involves an additional Physical Entity (the atmosphere) and in part because scheme-independent Measures exist that are very performance oriented (for example a decibel meter). Some subjects (for example, "Soil Investigation") do not appear to be related to any Limit States. Thus the relations between Limit States and THINGS are developed as the organizational Tree is constructed, and they may cause expansion and revision of the classification of Limit States.

The evaluation of some Limit States is particularly dependent on design and analysis methods and assumptions. Thus to aid the correct use of the performance criteria, procedural criteria for the techniques and parameters are to be formulated. The facet "Loads" is one that is particularly useful for the procedural criteria governing the analysis of the Limit States "in solid media." Provisions defining the "Loads" are applicable to a wide range of THINGS and care needs to be taken to avoid needless repetition of similar provisions.

Only a few Measures are included in the first round of classification, primarily to illustrate how the technique can move beyond performance criteria. The "Mechanical Measures" are typical of REQUIRED QUALITIES for performance oriented procedural criteria and the "Geometrical Measures" are typical of the REQUIRED QUALITIES for prescriptive criteria.

These characteristics of the Limit States and associated classifiers can be summarized by the following:

1) Limit States, and therefore performance criteria, are sometimes relatively independent of THING (for example "Noise") and other times dependent on the specific THING (for example "Instability").

2) Some Limit States are typically evaluated empirically (for example "noise") while others are analytically evaluated (for example "Instability"). Note that this is not equivalent to distinguishing between Limit States that are defined as the occurrence of a Phenomenon and Limit States that are defined as a certain magnitude of a Phenomenon. Deflection is an example of the second type that is often evaluated analytically.

3) Some procedural criteria are relatively independent of THING (for example provisions defining "Loads") while others are dependent on the THING (for example provisions for the strength of a "Vertical Cut" in "Clay").

4) All prescriptive criteria are dependent on the THING.

This indicates that following is an appropriate division for the criteria of the standard:

1) General Criteria

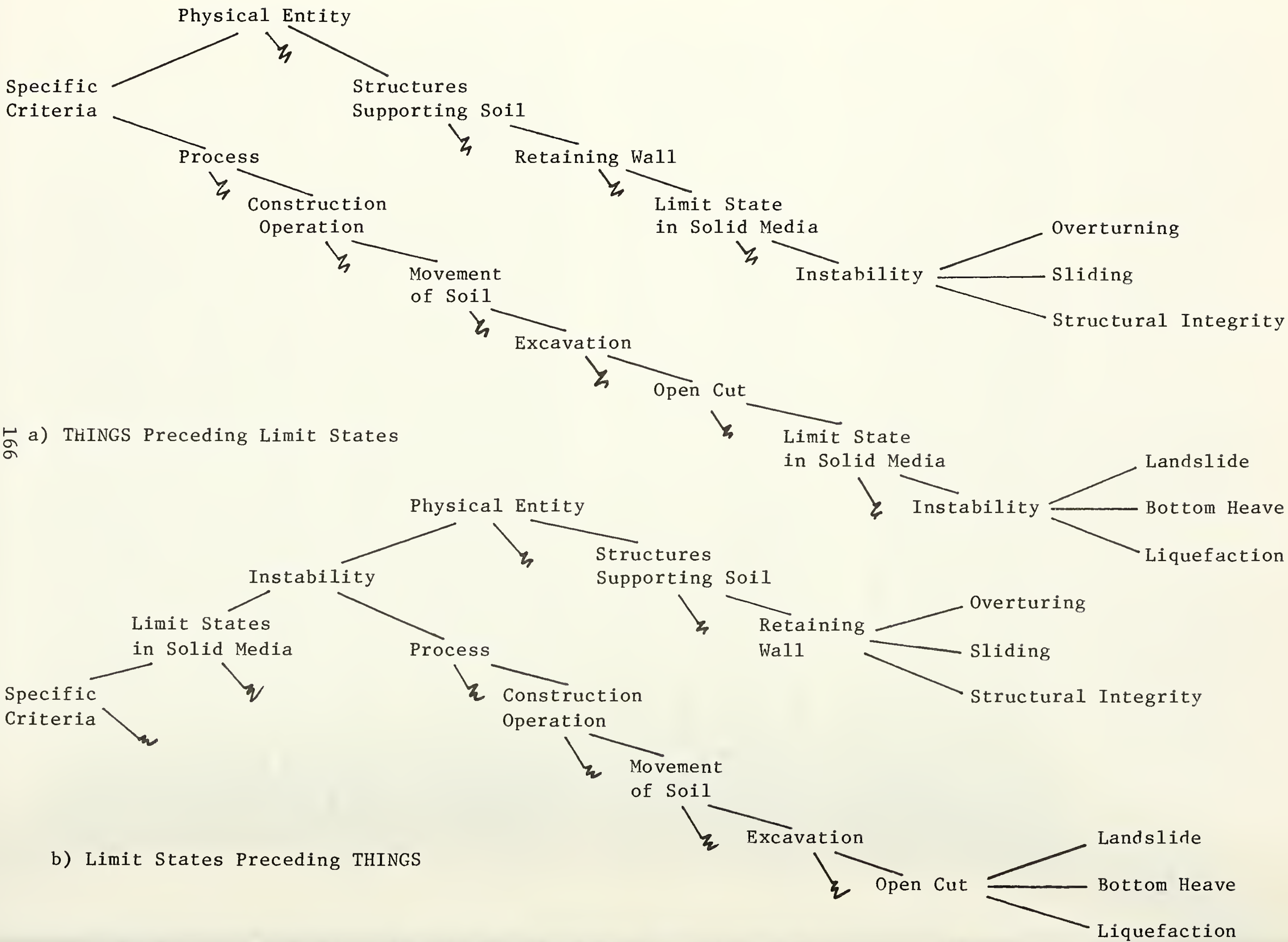
   - Performance Criteria
   - Procedural Criteria for Analytical Evaluation

2) Specific Criteria

   - Performance Criteria
   - Procedural Criteria for Analytical Evaluation
   - Prescriptive Criteria

The approach adopted for locating criteria in the two divisions is to initially assume that all are in the second part and then move those criteria frequently repeated in the organizational tree to the first part. Figure 5.2 shows a few selected branches of the organizational tree developed for the specific performance criteria. Note that the identification of a criterion does not depend on the hierarchical ordering of THING and Limit State. Each part of the figure identifies the same two criteria: one branch has the Classifiers "Retaining Wall," "Stability," and "Overturning" (neglecting the more general terms), while another has the classifiers "Excavation," "Open Cut," "Instability," and "Landslide."

Once the performance criteria are identified, the appropriate procedural and prescriptive criteria can be developed. This can be a simple extension of the organizational tree if some format is adopted. In this example it was initially assumed that there would be procedural criterion and a prescriptive criterion for each performance criterion and a nuclear tree with those two nodes is appended to the termination of each branch in the initial organizational tree. Figure 5.3 shows such an extension for one performance criterion. Thus the three level approach originally desired could result in a performance criterion stating the maximum acceptable risk of "Bottom Heave" in the excavation of an open cut, a procedural criterion for determining the "Strength" necessary to resist



a) THINGS Preceding Limit States

b) Limit States Preceding THINGS

Figure 5.2. Branches from Example Organizational Trees for Foundation Standard

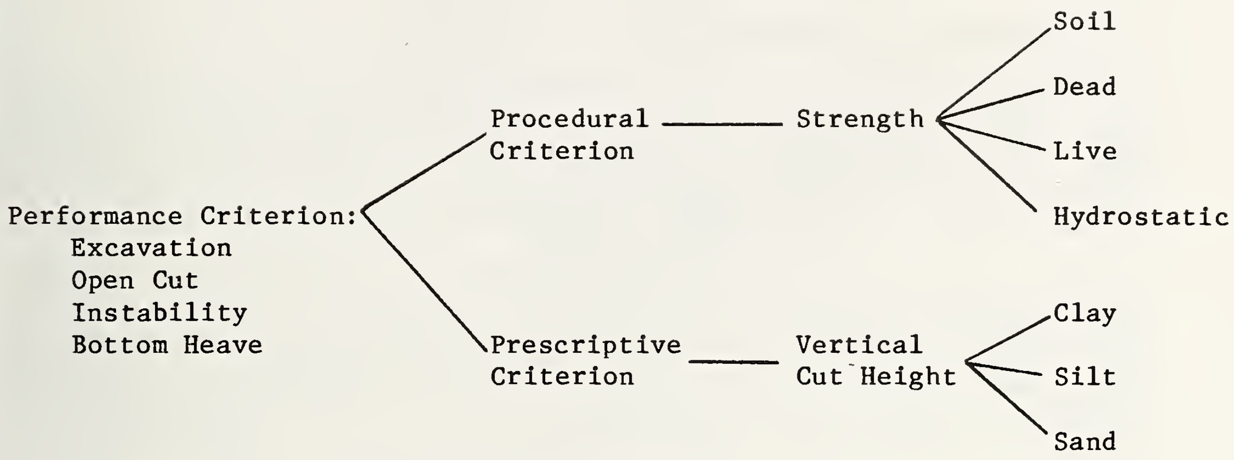


Figure 5.3 Extension of Organizational Tree (Foundation Standard)

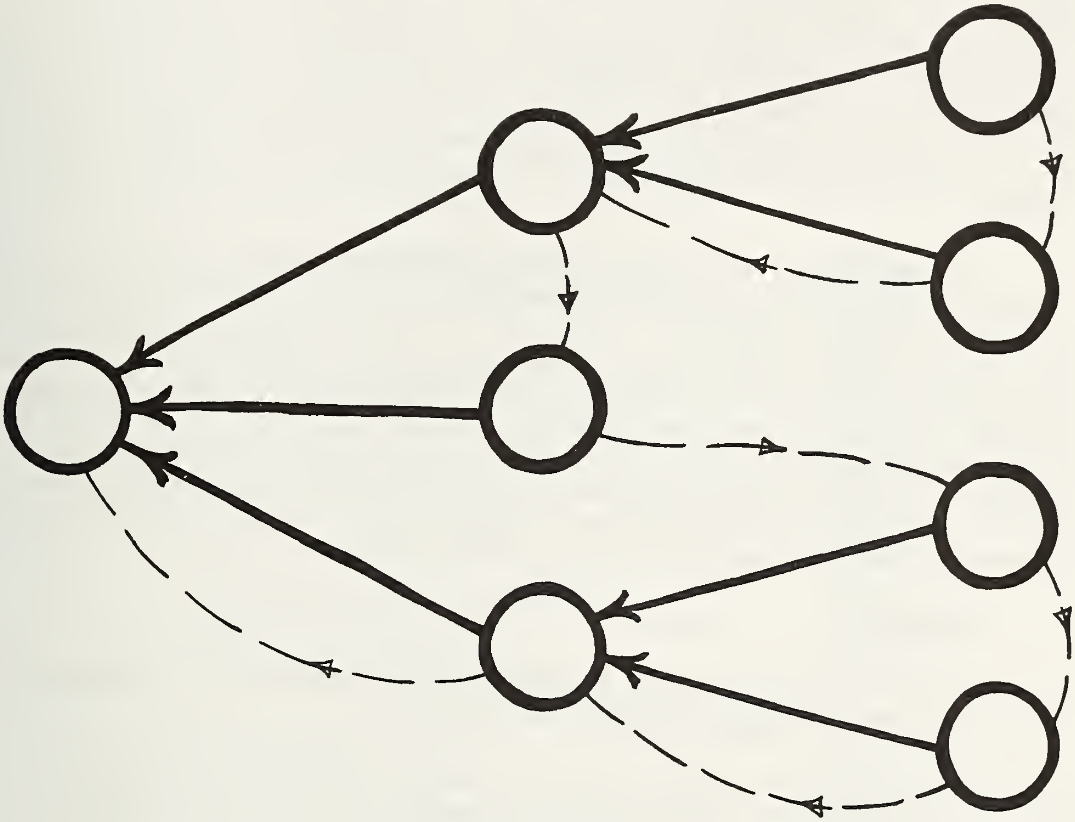
the effects of the various "Loads" and prescriptive criteria for the maximum "Height of Cut" of "Sideslope Ratio" for various soil types.

5.4 Comments on the Technique for Formulation

Establishng the scope of a standard and identifying the basic Requirements are extremely important tasks in the formulation of a standard. The technique described here can be a real aid in the process. It should be emphasized that the examples presented cannot effectively convey the necessity of iteration in the successful use of the technique.

The technique has the capability to extend beyond performance provisions, but performance concepts form the basis for sound organization. The technique can be used to focus deliberation on the performance basis of a standard before spending time on specific procedural or prescriptive provisions. This should promote efficiency and quality. The technique is also adaptable to a variety of formats.

Facing page: "Direct order" provides a bottom-up sequence; re-ordering an existing standard is a bottom-up procedure.



CHAPTER 6

EXPRESSION OF THE ORGANIZATION

Five means for finding provisions within a standard were described in section 1.2:

1) the table of contents

2) the index

3) headings that are printed within the text

4) proximity of related provisions

5) cross references written in the text

The system of classification for a given set of provisions is the basis for both the development of an index, as described in section 6.1, and the generation of outlines as described in section 6.2. The procedure for the generation of outlines is a systematic way of developing the first, third, and fourth means listed above. The information network described in section 2.3 also provides a means for ordering those provisions that are related by precedence and directly provides cross references between such provisions should they be separated by the outlining process. The use of the information network for ordering provisions and the relation between such use and the use of the outlining process for ordering are both discussed in section 6.3.

The generation of indexes and outlines is facilitated by the use of computer aids. The conceptual bases of several algorithms for such aids are presented in this chapter. A prototype computer program has been written that incorporates several algorithms developed in this study. (Chapter 2 references the existing outlining techniques and computer progarams.) Results from the prototype program are used freely in this chapter. The prototype program is partially documented in the appendix. Full documentation is not justified because a more sophisticated general purpose program is currently being developed [34, 35] and because the prototype is a research tool not suitable for production use.

6.1 Indexing

In describing subject indexes for libraries, Vickery states the two most important criteria for an index [130]:

> . . . the headings must be related in the mind of the searcher to the subject he seeks, and so constructed or arranged so that any particular heading can be located.

His first criterion is satisfied through the use of classifiers for the headings provided that the classification system is Relevant and Meaningful in the sense explored in chapters 3 and 4. His second criterion is most easily satisfied through the use of an alphabetical ordering of the headings. In some situations it may be wise to supplement the index with the display of the classification system alone in order to, once again in the words of Vickery [130], ". . . reveal the relations existing between the headings." This is most appropriate for classification systems for which the relations would not be apparent in the alphabetical list.

6.1.1 Form and Content of Indexes

An index is a list of headings with a list of references to provisions associated with each heading that provides access to the provisions. The simplest form is an alphabetical arrangement of the headings in a single level list with a cluster of references to the provisions associated with each heading given below or beside the heading (hereafter called a "simple index"). Table 6.1 is an example of such an index; it is a portion of the

Table 6.1 Example from a Simple Index

```
ALPHABETICAL INDEX

                                                                          REFERENCE LOCATION

ACCESS/EGRESS BLOCKED
     1472 GROUP III ACCESS REQUIREMENT                                    1.4.2(E)
ALTERATION
     1380 ALTERATION AND REPAIR REQUIREMENT                               1.3.2
ANALYSIS
     3105 STRUCTURAL ANALYSIS REQUIREMENT                                 3.1
     3381 CATEGORY C AND D INTERACTION REQUIREMENT                        3.3.4(B)
     4001 EQUIVALENT LATERAL FORCE ANALYSIS REQUIREMENT                   CHAPTER 4
     5001 MODAL ANALYSIS REQUIREMENT                                      CHAPTER 5
     5210 MODELING REQUIREMENT                                            5.2
     5310 MODES REQUIREMENT                                               5.3
     5410 PERIOD AND MODE SHAPE ANALYSIS REQUIREMENT                      5.4
     6001 SOIL STRUCTURE INTERACTION ANALYSIS REQUIREMENT                 CHAPTER 6
ANALYTICAL EVALUATION
    13226 ANALYTICAL EVALUATION PROCEDURES REQUIREMENT                    13.2.2
    13228 ANALYSIS METHOD REQUIREMENT                                     13.2.2
    13230 DETAILS OF ANALYTICAL EVALUATION REPORT REQUIREMENT             13.2.2
    13246 RESULTS OF ANALYTICAL EVALUATION                                13.2.2
    13248 GOVERNING EARTHQUAKE CAPACITY RATIO                             13.2.2
    13262 ALLOWABLE EARTHQUAKE CAPACITY RATIO                             13.2.2
ANCHORAGE
     8165 A/M/E ATTACHMENT REQUIREMENT                                    8.1.2
     8240 EXTERIOR WALL PANEL ATTACHMENT REQUIREMENT                      8.2.3
     8315 AMPLIFICATION FACTOR FOR ATTACHMENT OF M/E COMPONENT            8.3.2(A)
     8321 TYPE OF RESILIENT MOUNTING SYSTEM                               8.3.3, 2.1
     8345 MECH/ELEC ATTACHMENT DESIGN REQUIREMENT                         8.3.3
     8369 M/E ATTACHMENT CERTIFICATION (TESTING) REQUIRED                 8.3.4
ARCHITECTURAL
     8100 ARCHITECTURAL/MECHANICAL/ELECTRICAL PROVISIONS APPLICABLE       8.1
     8105 A/M/E PERFORMANCE LEVEL                                         8.1, 8.1.3
     8106 PERFORMANCE LEVEL FROM TABLE8-B                                 TABLE 8-B
     8190 PERFORMANCE CHARACTERISTIC FACTOR                               8.1.3, TABLE 8-A
     8200 ARCHITECTURAL DESIGN REQUIREMENT                                8.2.1, 8.2.3, 8.2.4, 8.2.5
     8215 SEISMIC FORCE FOR ARCHITECTURAL COMPONENTS                      8.2.2
     8220 SEISMIC COEFFICIENT FOR ARCHITECTURAL COMPONENTS                8.2.2, TABLE 8-B
     8240 EXTERIOR WALL PANEL ATTACHMENT REQUIREMENT                      8.2.3
     8250 ARCHITECTURAL COMPONENT DEFORMATION REQUIREMENT                 8.2.4
     8270 ARCH COMPONENT OUT OF PLANE BENDING REQUIREMENT                 8.2.5
AXIAL STRESS
    11862 CAT C AND D CONCRETE BOUNDARY MEMBER AXIAL STRENGTH REQT        11.8.4
    12754 MASONRY SHEAR WALL COMPRESSION STRESS REQUIREMENT               12.7.3
BEAM
    11602 ORDINARY CONCRETE FLEXURAL MEMBER REQUIREMENT                   11.6.1
    11604 ORDINARY CONCRETE FLEXURAL MEMBER REINFORCEMENT REQUIREMENT     11.6.1
    11618 ORDINARY CONCRETE FLEXURAL MEMBER MOMENT RESISTANCE REQT        11.6.1
    11628 ORDINARY CONCRETE FLEXURAL MEMBER REINFORCEMENT ANCHORAGE       11.6.1
    11640 ORDINARY CONCRETE FLEXURAL MEMBER WEB REINF REQUIREMENT         11.6.1
    11708 SPECIAL CONCRETE FLEXURAL MEMBER REQUIREMENT                    11.7.1
    11710 SPECIAL CONCRETE FLEXURAL MEMBER PROPORTIONING REQT             11.7.1
    11716 SPECIAL CONCRETE FLEXURAL MEMBER REINFORCEMENT REQT             11.7.1(A)
    11719 SPECIAL CONCRETE FLEXURAL MEMBER REINFORCEMENT SPLICE REQT      11.7.1(A)
    11732 SPECIAL CONCRETE FLEXURAL MEMBER LATERAL REINFORCEMENT REQT     11.7.1(B)
    11734 SPECIAL CONCRETE FLEXURAL MEMBER DESIGN SHEAR REQT              11.7.1(B)
```

index produced for the new seismic design provisions [50]. Each reference in a cluster consists of three parts: the datum number, the descriptive title of the datum, and the section or chapter number of the original text containing the provision. The datum number alone might suffice for access in some instances, but the descriptive title is particularly helpful in a simple index. It would be possible to include the page number or even the line number. Note that the most common index for a book is a single level list of headings with a cluster of page numbers for each heading.

The characteristic defect of simple indexes is that many headings reference clusters containing too many provisions for efficient use. The principle reason for development of indexes with multiple levels of headings, such as in table 6.2, is to subdivide the clusters of provision references into intelligible groups (say 5 to 10, as suggested in [87]).

Indexing principles can be formulated for indexes with multiple levels of headings and with several classifiers for each heading (see table 6.2 for examples of such indexes). The former type is quite useful for large indexes and will be discussed in more detail in section 6.1.3. The latter type introduces more power and possibly more relevance, but also causes additional problems of lengthiness. It does not seem worth the added cost when the provision reference in the index contains the datum description because the description and a heading containing most or all of the arguments of the provision are frequently very similar.

It is useful to include both Requirements and Determinations in the index. Even though all terminal datums correspond to Requirements, use of the standard frequently requires access to non-terminal provisions for many purposes. This is particularly true for Determinations that determine important characteristic quantities. The outline system primarily, but not wholly, provides access to the Requirements, and the information network system primarily provides access to Determinations through the ordering of the global ingredience of Requirements. The index is essentially a backup access system and must provide access for both types of provisions.

6.1.2 Classification Applied to Indexing

As described in section 4.2.2, the logical principles for classing a provision may be violated for the purpose of indexing. Generally speaking, the arguments of a provision for outlining will be fewer in number than those for indexing. A common example would be a Requirement with compound THINGS or REQUIRED QUALITIES (they occur, even though they are less than desirable, as discussed in section 4.1.3). Such a provision must be related to a general classifier for unique placement in an outline, but for reliable access through an index it would be better if the provision were related to specific classifiers for each part of the compound. Another common example is the addition of more general classifiers for indexing when the arguments are of a very narrow scope. Thus a provision classed as "Equivalent Lateral Force" from the classification of table 4.21b might also be classed as "Seismic Load Analysis" for indexing.

Table 6.2 Examples of Advanced Indexes

a) Multiple Level Headings

```
CAST-IN-PLACE
     11668 Minimum Distance for Lateral Reinforcement
     11680 Maximum Allowable Spacing of Lateral Reinforcement
     11743 Location Requires Hoop Reinforcement
     11770 Minimum Distance for Special Lateral Reinforcement
CATEGORY A
      3620 Category A Design and Detailing Requirement
      9300 Category A Wood Requirement
     11300 Category A Concrete Requirement
     11310 Category A Concrete Framing Requirement
     11340 Category A Concrete Anchor Bolt Requirement
CATEGORY B
   BEAM
        11602 Ordinary Concrete Flexural Member Requirement
        11604 Ordinary Concrete Flexural Member Reinforcement Requirement
        11618 Ordinary Concrete Flexural Member Moment Resistance Req't
        11628 Ordinary Concrete Flexural Member Reinforcement Anchorage Req't
        11640 Ordinary Concrete Flexural Member Web Reinf Requirement
   CASED
         7476 Category B Cased Concrete Pile Requirement
   COLUMN
        11662 Ordinary Concrete Beam Column Lateral Reinforcement Req't
   COMPONENT
      CASED
            7476 Category B Cased Concrete Pile Requirement
      CONCRETE
            7452 Category B Uncased Concrete Pile Requirement
            7476 Category B Cased Concrete Pile Requirement
            7490 Category B Concrete Filled Steel Pipe Pile Requirement
            7492 Category B Precast Concrete Pile Requirement
            7494 Category B Prestressed Concrete Pile Requirement
      DETAILED DESIGN
            3630 Category B Design and Detailing Requirement
            3640 Category B Openings Requirement
```

b) Multiple Classifier Headings

```
   CATEGORY B  CONCRETE ORDINARY MOMENT FRAME
     11600 Category B Concrete Ordinary Moment Frame Requirement
     11602 Ordinary Concrete Flexural Member Requirement
     11604 Ordinary Concrete Flexural Member Reinforcement Requirement
     11618 Ordinary Concrete Flexural Member Moment Resistance Req't
     11628 Ordinary Concrete Flexural Member Reinforcement Anchorage Req't
     11640 Ordinary Concrete Flexural Member Web Reinf Requirement

   CATEGORY B  CONCRETE PILE REINFORCEMENT
      7452 Category B Uncased Concrete Pile Requirement
      7476 Category B Cased Concrete Pile Requirement
      7490 Category B Concrete Filled Steel Pipe Pile Requirement
      7492 Category B Precast Concrete Pile Requirement
      7494 Category B Prestressed Concrete Pile Requirement
```

Some classifiers that are quite useful in outlining are associated with too many provisions to be useful for indexing. Consider the two classes "Material Generic" and "Material Specific" from the classification in table 4.21a. Nearly all of the 405 provisions classified in that study were classed as one or the other. The two classes provide a significant division between two major portions of the standard useful in outlining, but their inclusion in an index simply increases the length of the index without adding to its usefulness. Thus, such classifiers should not be used as headings in an index.

In some instances classifiers are associated with provisions purely for the purpose of ordering in outlining. For example, Requirements for the performance of a structure might be classed according to the stages of the design process in which the Requirement normally would be satisfied. Such classification is relevant in outlining, but possibly can be misleading in indexing if the same classifiers are used to indicate the THING or REQUIRED QUALITY of some Requirements and used for arranging other Requirements pertaining to other THINGS or REQUIRED QUALITIES. The user of the index has no sure way of distinguishing between these two purposes, and he would probably assume that the heading (classifier) is related to the THING for each associated provision. For the purpose of indexing, a provision should be deleted from the scope list of a classifier when that classifier is associated with the provision only for purposes of arrangement.

Logical classing is not necessary for indexing because the structure of an index depends primarily on the relation of a classifier and a provision. The relations between classifiers are much less important. The index does not have a unique location for each provision reference; it generally will have several, even if the classing is strictly logical. The index benefits from relaxation of logical rigor when it can make the product more natural, and when the relaxation can do no harm.

The literature indicates that an alphabetical ordering of the headings in an index is meaningful and well accepted. The other choice for indexing is to order the headings by their positions in the trees of classifiers. The alphabetical order appears preferable because it is common to so many indexes and because doing so relieves one from making decisions about the ordering of fields and facets for use as an index. Adopting the alphabetical order adds importance to the ordering of words in multi-word headings. Thus classifiers containing more than one word ideally should have the most relevant word placed first. Headings composed of more than one classifier have not been pursued in any detail in this study, but should they be used, the order of terms suggested by Vickery and shown in figure 3.2 is worthy of consideration.

6.1.3 Algorithms for Index Generation

The production of an alphabetical index of the simple type by a computer is quite elementary, once the classification, the provision references, and the arguments for each provision have been stored. Two preliminary steps are the transposition of the argument lists to determine the provisions associated with each classifier and the alphabetical sorting of the classifier names. The only decision making necessary during the generation of the index is to suppress those classifiers associated with no provisions (common for very general classifiers), to delete those classifiers associated with too many provisions to be useful, and to delete provisions from the scope list of a classifier when the association is for purposes of arrangement only.

The production of multiple level indexes requires the subdivision of the clusters of provision references. One "brute force" way of subdividing a cluster is through the use of an alphabetical list of all classifiers remaining. Table 6.2a is such an index. The cluster associated with "Category B" is subdivided by the series of headings "Beam", "Cased", etc. The cluster associated with both "Category B" and "Component" is further subdivided. Such a method is simple to implement in a computer program but has a serious defect in that it may greatly increase the length of the index. Since a provision should be referenced each time it is relevant to the heading, a provision cited once in the single level cluster would appear as many times as one of its other arguments is used for a subheading. Note that datum 7476 appears on two levels in the subdivision of the cluster for "Category B" in figure 6.2a. In the study of the new seismic design provisions, the length of an index was increased six-fold by reducing the maximum cluster size from twenty-five to seven.

An ideal subdivision occurs if the clusters are subdivided so that each provision reference occurs in one and only one of the subdivisions. This, however, is possible only when the logical principles are followed in classing the provisions. Since there are good reasons for straying from strictly logical classing, particularly for indexing, a need exists for a usable, albeit not ideal, solution to the problem. The following scheme is offered as being ideal when the classification for the cluster is logical and minimally redundant when it is not (the scheme has not been implemented in the computer program described in the appendix):

1) Select one facet, not containing the heading in question, that is exhaustive (that is, all provisions are classed by it) to use as headings to create the next level of subdivision. Because it references all provisions, each provision of the cluster will be in at least one of the subdivisions. And because large scale violation of the logical principles is uncommon, few provisions will be in more than one of the subdivisions.

2) If no facet is exhaustive for all provisions, determine if any facet is exhaustive for the provisions in the cluster; if one is, use it as described in step 1. This would involve a brief analysis each time a cluster is to be subdivided, thus it is not preferred over step 1.

3) If no facet is exhaustive in any sense, select a multifaceted field, not containing the heading in question,

that is exhaustive for all provisions. Failing that, select a field that is exhaustive for the cluster. The likelihood that some provisions from the cluster will be repeated in the subdivisions is greatly increased because there is no logical rule preventing a provision from being classed according to more than one facet in a field.

4) If no field is exhaustive in any sense, or perhaps in lieu of step 3, use a facet (or field) that is not exhaustive, placing the references to those provisions not included in any of the subdivisions immediately beneath the original heading as a "general" group, preceding the first subdivision.

Such a scheme may be applied recursively to generate as many levels as necessary to lower the size of clusters to a desired maximum size, except that the fourth option in the scheme does not present any way to divide the "general" subcluster. The recursive application requires that the facets of the active headings at each level above be avoided when selecting a new facet for additional subdivision.

6.2 Outlining

The goal for the process of outlining is to find the best linear order of the provisions - the order that maximizes the desired qualities of organization. The approach presented here involves two activities:

1) Generating alternative outlines with strategies that promote the desirable qualities

2) Measuring the qualities of different outlines to compare their overall goodness for the specific intended use.

The generation of outlines is discussed in 6.2.1, with more explicit detail added in 6.2.2 and 6.2.3. Much of the basis for preference of one outline over another is individual and subjective. The approach has the advantage of being able to provide different outlines of the same provisions for different users. A discussion of several useful measures is contained in 6.2.4.

Most of the examples in this section are taken from the analysis of the new seismic design provisions [50]. One of the findings pointed out in chapter 4 should be recalled: that analysis included many unnecessary Requirement datums. The effects of reformulating those datums upon the outline examples presented here are postulated and pointed out where they are thought to be pertinent.

6.2.1 Generation of Outlines

The generation of an outline from a classification for a set of provisions is similar to the technique described for the formulation of a standard in chapter 5. First, an organizational tree is generated by appending nuclear trees of classifiers together, then provisions are entered on the branches of the tree according to their arguments. Compared to the technique for formulation, the restrictions on the selection of trees that must be included (such as THING and REQUIRED QUALITY) or their order (such as Performance Attribute-Phenomenon-Limit State, etc.) are relaxed because the outline only serves to order existing provisions, not to identify potential provisions. For the same reason, the organizational tree need not be a complete representation of the classification. It is conceivable that a single facet could provide a useful ordering of a set of provisions. Outlining is subject to some requirements that formulation is not, such as the necessity of providing a location for each existing provision.

The portion of the new seismic design provisions dealing with quality assurance provides a simple example for an overview of outlining. Table 6.3 shows the provisions and their arguments, which are a subset of the classifiers in table 4.21. Note that "Building" is an argument of each of the Requirements in table 6.3, but none of the Determinations. A rather arbitrary rule was followed in that study, which required that each Requirement be associated with a Physical Entity classifier, even if the subject of the Requirement was a Building Process. The rule was useful in providing an overall outline according to Physical Entity, but it results in an apparent distinction between Requirements and Determinations that is not relevant in arranging this subset of the provisions.

There are two trees of classifiers in table 6.3 that are relevant for arrangement: "Quality Assurance" and "Social Qualities." Outlines can be produced using either one independently; see table 6.4 for such an outline based on the tree quality assurance. Outlines can also be produced by appending either tree on to the other; see table 6.5 for an example that uses the tree "Social Qualities" for the major subdivision and the tree "Quality Assurance" for the minor subdivisions. It is frequently necessary or desirable to reduce the hierarchial depth for an outline by a process of condensing headings as shown in table 5.7. Such a process would delete one of the two levels in table 6.5 for the classifiers "Method" and "Technique," because that extra level is not necessary for the proper grouping of the provisions.

Outlining differs from indexing and from formulation in the necessity for logical rigor. Because an outline is intended to be a single point of access tool, logical unambiguity is more important than it is for indexing, in which classing a provision by two siblings is acceptable. (Note that the outlining process does not guarantee that a provision will appear only once in the outline; the problem is discussed in the following section.) And because an outline is an access tool, logical completeness is not as important as it is for formulation, which is an identification tool. The outline of table 6.4 may suffice for access, but certainly would not serve to identify the scope of the provisions. The outline need only be complete enough to include the provisions, not to identify them.

TABLE 6.3 Classification of Quality Assurance Provisions

Provisions	Classifiers															
	Building	Mechanical/Electrical	Equipment	Construction	Quality Assurance	Planning	Inspection	Testing	Social Qualities	Existence of Process	Method	Technique	Principles &	Assumptions	Documentation	Scope
1601 Quality assurance k	X				X				X							
1602 Quality assurance plan required (D)					/	X				X						X
1604 Quality assurance plan acceptance R	X				/	X									X	
1605 Details of quality assurance plan R	X				/	X						X				
1613 Statement of contractor on QA plan R	X			/	/	X									X	
1625 Quality assurance personnel R	X				X				X							
1628 Minimum special inspection (D)					/		X					X				X
1635 Minimum special testing (D)					/			X				X				X
1637 Mech/elect equip. testing required (D)		/	/		/			X		X						X
1640 Mech/elect testing plan acceptance R	X	/	/		X										X	
1641 Min. special testing for Mech/elect (D)		/	/		/			X				X				X
1644 Mech/elect test compliance R	X	/	/		/			X				X				
1651 Quality assurance plan compliance R	X				X							X				
1654 Quality assurance reporting R	X				X										X	
1655 Special inspectors weekly report R	X				/		X								X	
1662 Special inspectors final report R	X				/		X								X	
1668 Contractors final report R	X			/	X										X	
1674 Mech/elect equip mfg certification R	X	/	/		/			X							X	

NOTE: The "X" indicates an argument for outlining and indexing, while the "/" indicates an argument for indexing only. The "D" indicates a Determination.

Table 6.4 Outline for Quality Assurance Provisions Based on Process Tree

CLASSIFIER	PROVISIONS	
QUALITY ASSURANCE	1601	QUALITY ASSURANCE REQUIREMENT
	1625	QUALITY ASSURANCE PERSONNEL ARRANGEMENTS
	1640	MECHANICAL/ELECTRICAL TESTING PLAN ACCEPTANCE REQUIREMENT
	1651	QUALITY ASSURANCE PLAN COMPLIANCE REQUIREMENT
	1654	QUALITY ASSURANCE REPORTING REQUIREMENT
	1668	CONTRACTORS' FINAL REPORT REQUIREMENT
PLANNING (QA)	-1602	QUALITY ASSURANCE PLAN REQUIRED
	1604	QUALITY ASSURANCE PLAN ACCEPTANCE REQUIREMENT
	1605	DETAILS OF QUALITY ASSURANCE PLAN
	1613	STATEMENT OF CONTRACTOR ON QUALITY ASSURANCE PLAN
INSPECTION	-1628	MINIMUM SPECIAL INSPECTION
	1655	SPECIAL INSPECTORS' WEEKLY REPORT REQUIREMENT
	1662	SPECIAL INSPECTORS' FINAL REPORT REQUIREMENT
TESTING	-1635	MINIMUM SPECIAL TESTING
	-1637	MECHANICAL/ELECTRICAL EQUIPMENT TESTING REQUIRED
	-1641	MINIMUM SPECIAL TESTING FOR MECH/ELECT EQUIPMENT
	1644	MECHANICAL/ELECTRICAL TEST COMPLIANCE REQUIREMENT
	1674	MECH/ELECT EQUIP MANUFACTURER CERTIFICATION PROGRAM REQT

ALL PROVISIONS WERE OUTLINED

NO PROVISIONS WERE OUTLINED MORE THAN ONCE

Table 6.5 Outline for Quality Assurance Provisions Based on Appending Process Tree to Required Quality Tree

```
CLASSIFIER                                                  PROVISIONS


 BUILDING
   REQUIRED QUALITIES
     BUILDING PROCESSES
       QUALITY ASSURANCE
         SOCIAL QUALITIES ....................................  1601  QUALITY ASSURANCE REQUIREMENT
                                                                1625  QUALITY ASSURANCE PERSONNEL ARRANGEMENTS
           EXISTENCE OF PROCESS
             PLANNING (QA) ...................................  -1602  QUALITY ASSURANCE PLAN REQUIRED
             TESTING .........................................  -1637  MECHANICAL/ELECTRICAL EQUIPMENT TESTING REQUIRED
           METHOD
             TECHNIQUE .......................................  1651  QUALITY ASSURANCE PLAN COMPLIANCE REQUIREMENT
               PLANNING (QA) .................................  1605  DETAILS OF QUALITY ASSURANCE PLAN
               INSPECTION ....................................  -1628  MINIMUM SPECIAL INSPECTION
               TESTING .......................................  -1635  MINIMUM SPECIAL TESTING
                                                                -1641  MINIMUM SPECIAL TESTING FOR MECH/ELECT EQUIPMENT
                                                                1644  MECHANICAL/ELECTRICAL TEST COMPLIANCE REQUIREMENT
           DOCUMENTATION .....................................  1640  MECHANICAL/ELECTRICAL TESTING PLAN ACCEPTANCE REQUIREMENT
                                                                1654  QUALITY ASSURANCE REPORTING REQUIREMENT
                                                                1668  CONTRACTORS' FINAL REPORT REQUIREMENT
             PLANNING (QA) ...................................  1604  QUALITY ASSURANCE PLAN ACCEPTANCE REQUIREMENT
                                                                1613  STATEMENT OF CONTRACTOR ON QUALITY ASSURANCE PLAN
             INSPECTION ......................................  1655  SPECIAL INSPECTORS' WEEKLY REPORT REQUIREMENT
                                                                1662  SPECIAL INSPECTORS' FINAL REPORT REQUIREMENT
             TESTING .........................................  1674  MECH/ELECT EQUIP MANUFACTURER CERTIFICATION PROGRAM REQT


ALL PROVISIONS WERE OUTLINED


NO PROVISIONS WERE OUTLINED MORE THAN ONCE
```

The establishment of a priority ranking of arguments for each provision and use of the priority in outlining is described briefly in chapter 2. (See table 2.4 for a hypothetical example.) Use of priority in outlining creates problems, however. No relevant basis has been found for ranking THING ahead of the REQUIRED QUALITY, or vice versa, although it is possible to hypothesize a situation in which a ranking of arguments from two facets belonging to one field might be relevant. Recent analysis indicates that such was the case in the study in which the use of priority was introduced [51, 135]. However, in the study of the seismic design provisions [50], which is a much larger example, no situation was found in which prioritizing arguments was useful. The use of many different facets for descriptive qualities probably precluded any advantage for the use of priority. A second problem presented by the use of priority is that it tends to reduce the number of possible outlines that differ in substantive ways, thus restricting freedom of choice. It may be said that a preference for one outline over another is in part based on a feeling that the headings are properly graded, which is in effect preferring a certain order of classifiers on a branch. It is appropriate to delay making such a choice from the stage of classing a provision to the stage of comparing two outlines.

It is possible that a distinction between two arguments from different but equivalent fields would be of some use if either could be the THING or REQUIRED QUALITY. For example, such a distinction could be used in table 6.3 to indicate that "Building" is not the primary subject of the quality assurance requirements. Such a distinction could easily be of use in indexing, and its potential use as a priority for outlining should be investigated.

The most important problem with priority is the inclusion in an organizational tree of multiple branches with the same set of classifiers. The minor distinction of ordering is unlikely to prevent ambiguous interpretation by many readers; it is likely a reader will miss relevant provisions. The difference between a subsection on "Yielding" in a section on "Beams" and a subsection on "Beams" in a section on "Yielding" is not likely to be meaningful. Thus the use of priority in classification and outlining is not given further significant consideration in this study.

The techniques of creating outlines are discussed in more detail in the following two sections. In section 6.2.2 criteria are discussed for entry of a provision on a branch of an organizational tree, and in section 6.2.3 the methods for generating organizational trees are discussed. The inverted order of discussion is appropriate because the criteria for entry are independent of the method of generation, but some of the methods of generation require the use of criteria for entry as a part of the process.

6.2.2 Entry of Provisions in an Organizational Tree

Entry depends upon the comparison of the set of classifiers that compose a branch of an organizational tree with the set of outlining arguments for a provision. The comparison reveals whether the provision is appropriately identified by the classifiers on the branch. The decision is based on the objectives for the organization of a standard given in section 1.2, as follows:

1) Relevant: each classifier on the branch must be related to one of the provision's outlining arguments.

2) Complete: each outlining argument of the provision must be included among the set of classifiers on the branch.

3) Unique: no classifier on the branch may be a "cousin" (this concept is defined subsequently) of any outlining argument.

4) Graded: no classifier on the branch may be a descendant of any outlining argument.

The decision does not depend on the objectives Even, Minimal, and Progressive because they are not relevant in the context of a single branch of an organizational tree. Meaningful is achieved when the arguments are selected for a provision. It is possible to consider a limit on the number of nodes on a branch as a criterion for the objective Intelligible. Such a check is of more worth following the condensation of the hierarchical depth of an organizational tree as mentioned in the previous section and in table 5.7.

Each of the four criteria depend on the logic of the classification system. Since a faceted classification system need not be strictly logical, checking some of the criteria becomes complex. As mentioned in section 4.2.1, it is possible to combine all the facets in a field into a large tree of classifiers, but frequently it is also possible to combine the facets in a relevant fashion into a large tree that is not completely logical. The problem is to define clearly how the objectives can be attained with a less than perfectly logical classification system.

Given a faceted system, the criterion for Relevant is easily described in three steps. (Recall that each classifier on the branch must pass the criterion for the provision in question to be qualified for entry on the branch.)

1) A classifier that is an argument of the provision is relevant.

2) A classifier from the same facet as an argument is relevant if it is a logical predecessor of the argument. (A logical predecessor is the parent, the parent of the parent, etc.)

3) A classifier from the same field, but not the same facet, as an argument is relevant if it is a logical predecessor of the argument in a large tree of combined facets.

The second test can be extended to say that a classifier from the same facet as an argument is not relevant unless it is a predecessor of the argument (assuming it failed the first test). Making this extension means that the criteria for Unique and Graded mentioned previously are automatically included, as far as facets are concerned. As shown in figure 6.1, there are four possibilities for a classifier and an argument from the same facet: the classifier is a predecessor, descendent, or cousin of the argument or is the same as the argument. As shown in figure 6.1, a cousin includes the siblings, the siblings of the predecessors, and any decendents of such siblings. The third test for Relevant cannot be similarly extended, (that is, it cannot be used to disqualify a provision simply because one classifier fails the test), because the possibility of illogical siblings leaves the possibility that the classifier may be relevant for another argument for the same provision.

The principal problem in using these tests for Relevant is that a complete specification of the permissible interconnections of the facets is required as a part of the classification system. Otherwise a potentially relevant classifier that is not traceable to any argument will disqualify a provision from appearance on the appropriate branch. A system for recording and using relations between facets is included in the computer program described in the appendix. It shows promise, but further study is necessary. An added problem is that the tracing of the permissible interconnections among facets involves significantly more checking than any of the other tests employed for entry of a provision on a branch.

There is a less stringent criterion for relevant that frequently is good enough to produce useful outlines. The criterion combines well with the relatively simple criteria for Unique and Graded, which tend to prevent irrelevance in addition to delivering their named objectives. The criterion is quite simple: the last classifier on the branch must be an argument of the provision. It is accomplished efficiently by selecting those provisions in the scope list of the last classifier and discarding all others. Since the last classifier on the branch usually has a smaller scope list than the first classifiers on a branch, it also has the desirable feature of limiting the provisions to be checked by the other criteria in an intuitively optimal fashion. For convenience, this less stringent criterion is called "local relevance," while the previous criterion is called "full relevance." Useful outlines, such as that shown in table 6.4, may be obtained employing the local relevance criterion without any of the other criteria. Methods of synthesizing organizational trees, described in section 6.2.3, tend to promote the validity of the local relevance criterion.

The criterion for Complete assures that a provision is not prematurely entered into an outline. It has nothing to do with assuring that all provisions are outlined, another aspect of completeness. The criterion is independent of the question of relevance, and it is applied conveniently immediately following the local relevance criterion. For each provision passing the local relevance criterion, a check is made to see that each of its arguments is among the classifiers on the branch. The

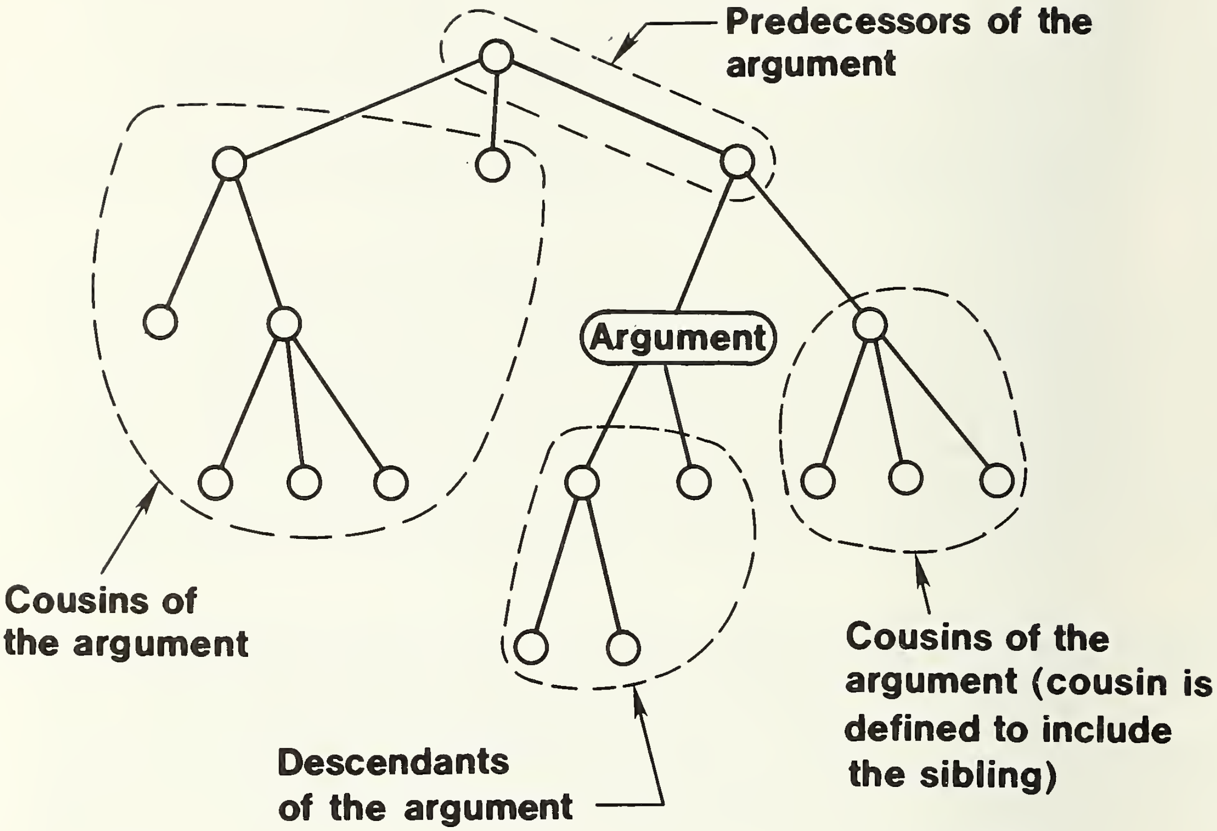

Predecessors of the argument
Argument
Cousins of the argument
Cousins of the argument (cousin is defined to include the sibling)
Descendants of the argument

Figure 6.1 Partition of a Tree into Logical Regions

outline of table 6.5 was produced by applying only the local relevance and completeness criteria.

A separate check for Unique at the facet level is often useful when the full relevance criterion is not employed. The criterion disqualifies any provision with an argument that is a cousin of one of the classifiers on the branch. Table 6.6a shows a portion of an outline generated in the analysis of the seismic provisions [50], and table 6.6b shows the same portion with a logical defect that prevents the entry of some important provisions when the criterion is applied. (The logical defect is that "Existence of Process," "Interrelationship," and "Quantities and Dimensions" cannot be descendents of "Strength Required.")

A separate check for Graded at the facet level often is useful when the full relevance criterion is not employed. The test is convenient to apply in conjunction with the Unique criterion just described, because of the nature of the task of partitioning a tree as shown in figure 6.1. Tables 6.7 and 6.8 show the impact of the graded criterion. They are outlines generated from the classification shown in table 6.3. Redundant, therefore ambiguous, locations for provisions may be a common defect in outlines developed without use of the graded criterion. Because logic is not preserved above the facet level is is not possible to apply criteria for Unique or Graded above that level.

A common defect in an outline that satisfies all the criteria except full relevance is the entry of a single provision on more than one branch of the organizational tree. The ambiguity usually arises from the use of a facet that is not exhaustive. (A facet is exhaustive if its combined scope list includes all provisions.) For example, consider a provision from the analysis of the seismic provisions with the arguments: "Part of Building," "Material Specific," and "Masonry." It would meet all criteria except full relevance for both the following branches of classifiers:

(i)	(ii)
Part of Building	Part of Building
Structural	Nonstructural
Material Specific	Material Specific
Masonry	Masonry

Structural and Nonstructural are not exhaustive; the provision could apply to either. Using the full relevance criterion, the provision would not qualify for either branch.

Even the full relevance criterion is not guaranteed to prevent all redundant entry of provisions, unless the fields are strictly logical. Consider the same provision just described and the following branches of classifiers:

(i)	(ii)
Part of Building	Part of Building
Material Specific	Material Specific
Component	Masonry
Masonry	

TABLE 6.6 Outline Showing Logical Defect Detected with Uniqueness Criterion

a) Correct organizational network

Classifiers	Provisions	
Foundation	7001	Foundation Design Requirements
Strength Required		
Soil	7230	Foundation Soil Capacity Reqt
Foundation Structure	7210	Foundation Component Strength Reqt
Category B	7400	Category B Foundation Requirement
Soil		
Existence of Process		
Site/Soil Investigation	7404	Category B Soil Investigation Reqt
Foundation Structure		
Interrelationship	7428	Category B Foundation Tie Reqt
Pile		
Quantities & Dimensions	7438	Category B Foundation Pile Reqt
Reinforcement (Concrete)		
Steel	7490	Category B Steel Pipe Pile Reqt

b) Incorrect organizational network

Classifiers	Provisions	
Foundation	7001	Foundation Design Requirements
Strength Required		
Soil	7230	Foundation Soil Capacity Reqt
Foundation Structure	7210	Foundation Component Strength Reqt
Category B	7400	Category B Foundation Requirement
Soil		
Existence of Process		
Site/Soil Investigation	(*)	
Foundation Structure		
Interrelationship	(*)	
Pile		
Quantities & Dimensions	(*)	
Reinforcement (Concrete)		
Steel	(*)	

* "Existence of Process," "Interrelationship," and "Quantities & Dimensions" do not pass the criterion (as arguments, not classifiers) because they are cousins of the classifier "Strength Required" (see table 4.21 for the classification).

Table 6.7 Alternate Outline of Quality Assurance Provisions without Graded Criterion

```
CLASSIFIER                                            PROVISIONS


 BUILDING
   REQUIRED QUALITIFS
     BUILDING PROCESSES
       QUALITY ASSURANCE
         SOCIAL QUALITIES ................................  1601  QUALITY ASSURANCE REQUIREMENT
                                                            1625  QUALITY ASSURANCE PERSONNEL ARRANGEMENTS
           PLANNING (QA)
             EXISTENCE OF PROCESS ......................... -1602  QUALITY ASSURANCE PLAN REQUIRED
             METHOD
               TECHNIQUE ..................................  1605  DETAILS OF QUALITY ASSURANCE PLAN
                                                            1651  QUALITY ASSURANCE PLAN COMPLIANCE REQUIREMENT
             DOCUMENTATION ................................  1604  QUALITY ASSURANCE PLAN ACCEPTANCE REQUIREMENT
                                                            1613  STATEMENT OF CONTRACTOR ON QUALITY ASSURANCE PLAN
                                                            1640  MECHANICAL/ELECTRICAL TESTING PLAN ACCEPTANCE REQUIREMENT
                                                            1654  QUALITY ASSURANCE REPORTING REQUIREMENT
                                                            1668  CONTRACTORS'FINAL REPORT REQUIREMENT
           INSPECTION
             METHOD
               TECHNIQUE .................................. -1628  MINIMUM SPECIAL INSPECTION
                                                            1651  QUALITY ASSURANCE PLAN COMPLIANCE REQUIREMENT
             DOCUMENTATION ................................  1640  MECHANICAL/ELECTRICAL TESTING PLAN ACCEPTANCE REQUIREMENT
                                                            1654  QUALITY ASSURANCE REPORTING REQUIREMENT
                                                            1655  SPECIAL INSPECTORS'WEEKLY REPORT REQUIREMENT
                                                            1662  SPECIAL INSPECTORS'FINAL REPORT REQUIREMENT
                                                            1668  CONTRACTORS'FINAL REPORT REQUIREMENT
           TESTING
             EXISTENCE OF PROCESS ......................... -1637  MECHANICAL/ELECTRICAL EQUIPMENT TESTING REQUIRED
             METHOD
               TECHNIQUE .................................. -1635  MINIMUM SPECIAL TESTINC
                                                           -1641  MINIMUM SPECIAL TESTING FOR MECH/ELECT EQUIPMENT
                                                            1644  MECHANICAL/ELECTRICAL TEST COMPLIANCE REQUIREMENT
                                                            1651  QUALITY ASSURANCE PLAN COMPLIANCE REQUIREMENT
             DOCUMENTATION ................................  1640  MECHANICAL/ELECTRICAL TESTING PLAN ACCEPTANCE REQUIREMENT
                                                            1654  QUALITY ASSURANCE REPORTING REQUIREMENT
                                                            1668  CONTRACTORS'FINAL REPORT REQUIREMENT
                                                            1674  MECH/ELECT EQUIP MANUFACTURER CERTIFICATION PROGRAM REQT

ALL PROVISIONS WERE OUTLINED

THE FOLLOWING PROVISIONS WERE OUTLINED THE INDICATED NUMBER OF TIMES


(3)  1640  MECHANICAL/ELECTRIAL TESTING PLAN ACCEPTANCE REQUIREMENT
(3)  1651  QUALITY ASSURANCE PLAN COMPLIANCE REQUIREMENT
(3)  1654  QUALITY ASSURANCE REPORTING REQUIREMENT
(3)  1668  CONTRACTORS'FINAL REPORT REQUIREMENT
```

Table 6.8 Alternate Outline of Quality Assurance Provisions with Graded Criterion

```
CLASSIFIERS                                                    PROVISIONS


  BUILDING
    REQUIRED QUALITIES
      BUILDING PROCESSES
        QUALITY ASSURANCE
          SOCIAL QUALITIES ..................................  1601  QUALITY ASSURANCE REQUIREMENT
                                                               1625  QUALITY ASSURANCE PERSONNEL ARRANGEMENTS
            PLANNING (QA)
              EXISTENCE OF PROCESS ........................... -1602  QUALITY ASSURANCE PLAN REQUIRED
              METHOD
                TECHNIQUE ...................................  1605  DETAILS OF QUALITY ASSURANCE PLAN
              DOCUMENTATION .................................  1604  QUALITY ASSURANCE PLAN ACCEPTANCE REQUIREMENT
                                                               1613  STATEMENT OF CONTRACTOR ON QUALITY ASSURANCE PLAN
            INSPECTION
              METHOD
                TECHNIQUE ................................... -1628  MINIMUM SPECIAL INSPECTION
              DOCUMENTATION .................................  1655  SPECIAL INSPECTORS'WEEKLY REPORT REQUIREMENT
                                                               1662  SPECIAL INSPECTORS'FINAL REPORT REQUIREMENT
            TESTING
              EXISTENCE OF PROCESS ........................... -1637  MECHANICAL/ELECTRICAL EQUIPMENT TESTING REQUIRED
              METHOD
                TECHNIQUE ................................... -1635  MINIMUM SPECIAL TESTING
                                                              -1641  MINIMUM SPECIAL TESTING FOR MECH/ELECT EQUIPMENT
                                                               1644  MECHANICAL/ELECTRICAL TEST COMPLIANCE REQUIREMENT
              DOCUMENTATION .................................  1674  MECH/ELECT EQUIP MANUFACTURER CERTIFICATION PROGRAM REQT


THE FOLLOWING PROVISIONS WERE NOT OUTLINED


     1640  MECHANICAL/ELECTRICAL TESTING PLAN ACCEPTANCE REQUIREMENT
     1651  QUALITY ASSURANCE PLAN COMPLIANCE REQUIREMENT
     1654  QUALITY ASSURANCE REPORTING REQUIREMENT
     1668  CONTRACTORS'FINAL REPORT REQUIREMENT


NO PROVISIONS WERE OUTLINED MORE THAN ONCE
```

"Component" belongs to a facet that can be used to distinguish among various "Parts of Buildings," and "Masonry" belongs to a facet that may be used to distinguish among various "Components." The former fact means that "Component" is not out of place in the branch, and the latter fact means that "Component" can be a predecessor of "Masonry," thus satisfying the full relevance criterion. Since "Component" is not an argument, the completeness criterion is satisfied for either branch. Thus the provision qualifies for both branches, an ambiguous situation.

Three possible solutions for this problem exist. First, the logical rules could be enforced for the entire classification, but for reasons already explained, fully logical classifications are not always desirable or possible.

Second, the full relevance criterion could be strengthened by requiring that each of the classifiers on the branch be an argument or the direct predecessor (within the same facet) of an argument. This would avoid tracing the relations between facets. It has the disadvantage of requiring added identification and storage of arguments. For example, consider the following arguments for a Requirement giving the minimum amount and spacing of reinforcement in a cased concrete pile:

Pile
Reinforced Concrete
Cased
Reinforcement
Quantities and Dimensions

Now consider the application of this extended criterion against the the following list of classifiers, representing an appropriate branch for the provision:

Building	- does not qualify
Required Quality	- direct predecessor of "Quantities and Dimensions"
Part of Building	- does not quality
Structural	- does not qualify
Foundation	- does not qualify
Foundation Structure	- direct predecessor of "Pile"
Pile	- is an argument
Quantities and Dimensions	- is an argument
Reinforcement	- is an argument
Reinforced Concrete	- is an argument
Cast-in-place	- direct predecessor of "Cased"
Cased	- is an argument

The addition of "Foundation" as an argument would quality "Foundation" and "Structure," and the addition of "Part of a Building" would qualify "Part" and "Building." For this provision, it is not a large problem to add two arguments, but it could be for others. Furthermore it seems redundant because the argument "Pile" clearly implies that all of those higher order classifiers are relevant.

The third possible solution is that a more complete and explicit set of rules is needed for tracing the relations among facets. One such test not included in the present system is to require that the argument indirectly related to the classifier must follow that classifier (not necessarily immediately) on the branch. Another is that the branch of classifiers contain at least one classifier from each facet involved in the linkage. While it is forseeable that some improvements on the system described in section 4.2.1 are possible, it does not seem likely that any such rule would prevent the redundancy introduced through the classifier "Component" in the previous example concerning the "Masonry" provision.

More study of the question of an appropriate system of tracing relations between facets for the purpose of checking relevance is in order. The need is not urgent because reasonably good results are obtained with the present system using the two criteria for Relevant.

Given that the relevance criterion is divided into two criteria, local and full, and that the full relevance criterion will not always be used, an efficient strategy for computer processing emerges:

1) Apply the local relevance criterion.

2) For each provision passing the local relevance criterion, test each argument for the Complete, Unique, and Graded criteria, in that order.

3) For each provision passing those criteria, test each of the set of classifiers for the full relevance criterion, disqualifying the provision if any classifier fails. The full relevance criterion is actually applied in three steps, repeated here for convenience: the classifier is relevant if it is an argument or if it is a logical predecessor of an argument, either within or outside the same facet.

Such an approach is included in the computer program described in the appendix. The application of the criteria is optional, in that testing may stop after any criterion, if desired.

A recommended strategy is to begin outlining using all the criteria, relaxing the application of the latter criteria if logical problems seem to preclude the development of outlines that can include all the provisions. The alternate strategy of beginning to outline without the more rigorous criteria, and resorting to them only if necessary to reduce the redundant occurances of provisions has the advantage of less costly computer processing, but it also has the disadvantage of letting irrelevant, and sometimes illogical, outlines be developed.

A useful feature for "fine tuning" an outline is the ability to selectively remove classifiers and arguments from the system. Table 6.3 shows that "Scope" is an argument for some of the provisions, yet it does not appear in the outlines of tables 6.5, 6.7, or 6.8, because in preparing those outlines the entire field "Derived Values" (table 4.21e) was dropped from consideration. Thus the completeness criterion could be satisfied without "Scope" appearing in the outline. It is also possible, and sometimes useful, to selectively ignore some of the entry criteria in various portions of an outline.

In summary, there are two independent criteria, for Relevant and Complete, that lead to five criteria that work in a faceted classification system: local relevance, Complete, Unique, Graded, and full relevance. It is possible to develop good outlines without using all the criteria, but ambiguities and inconsistencies are less likely when using all the criteria. Because it still is possible to develop outlines with more than one position for a provision when using all the criteria, more study of of the fifth criterion and of the logic of faceted systems is in order.

6.2.3 Generation of an Organizational Tree

The capability to generate alternative organizations for a set of provisions is one of the objectives of this study. The needs for a method with enough flexibility to provide several arrangements and enough rigor to preserve and promote the relevance and logic incorporated in the classification system might seem contradictory. None-the-less, such a capability has been the heart of previous studies, and is central in this one.

The initial problem in generating an organizational tree from a classification system is the selection of a working unit. In past studies the classification systems have been relatively small and uniform, and the entire tree for a field [51] or a level of such a field [97] have been used. With the expansion of the structure of the classification system described in section 4.2, it might be expected that the working unit would be a facet or a level of facet. However, neither has proven to be as good as the nuclear tree defined in section 4.2.1 and used for generating a tree in section 5.2. Recall that a nuclear tree is the smallest logical unit of a classification, preserving the basic concepts of mutual exclusion and collective exhaustion. The small unit allows great flexibility in the construction of organizational trees and the rules for correct use are not unduly complex.

It is convenient to discuss the rules for appending nuclear trees in the context of the objectives of organization. The objective Unique is preserved, assuming the original classification is mutually exclusive, by prohibiting the appending of more than one nuclear tree to any single node on the organizational tree. Violating this rule creates "step siblings", as shown in the following example derived from the analysis of the seismic provisions [50] (the classifiers are from table 4.21):

```
Foundation
  Soil
  Foundation Structure
    Strength ........................Requirement
    Interrelationship ...............Requirement
    Pile ............................Requirement
      Existance .....................Requirement
      Details .......................Requirement
```

The question created is whether piles are included in the strength and interrelation requirements for the foundation structure. Violation of this rule has been quite common in past models for organization. Considering the value of uniqueness to the user, this is an important objective for improvement.

The objective Complete is preserved if no siblings are dropped from an organizational tree. This rule is useful when comparing the scope of the provisions to the scope of the classification, which would be useful in the formulation of a standard. It is not really very necessary when the task is simply expressing an organization for a set of provisions. The objective Graded is preserved if a nuclear tree is never appended to a branch of an organizational tree that already incorporates a descendant of that nuclear tree. The rule is fundamental, but it can almost be taken for granted. Violations of it are unlikely to occur in most procedures for generating organizational networks.

The objective Progressive is preserved if siblings on a nuclear tree are not re-ordered. The objective Minimal provides an interesting strategy: to minimize the total number of headings in an organizational tree, append the nuclear trees with the larger number of siblings after those with the smaller number of siblings. This can be illustrated by an example involving three nuclear trees in which X_1, X_2 and X_3 represent the number of siblings on each tree in the order they are appended. In that case, the total number of nodes (headings), N, is:

$$N = X_1 + X_1X_2 + X_1X_2X_3$$

It may easily be shown for three trees that N is minimized if $X_1 \leq X_2 \leq X_3$. This expression for N can be generalized for n trees to:

$$N = \sum_{i=1}^{n} \prod_{j=1}^{i} X_j$$

It is also possible to generalize the rule for minimization: $X_1 \leq X_2 \leq \ldots X_n$. Note that this strategy minimizes the total number of nodes. The number of terminal nodes is the same for any order of appending the trees. It bears repeating that Minimal is a desirable, not a requisite property, and that other objectives take precedence.

Meaningful, Even, and Intelligible provide no rules for appending nuclear trees, but Relevant does. Perhaps the most interesting problem created by selecting the smallest possible working unit for creating organizational trees is that of assuring relevance. The problem has not arisen in previous studies. In exploring the issue, it is useful to distinguish between three types of nuclear trees:

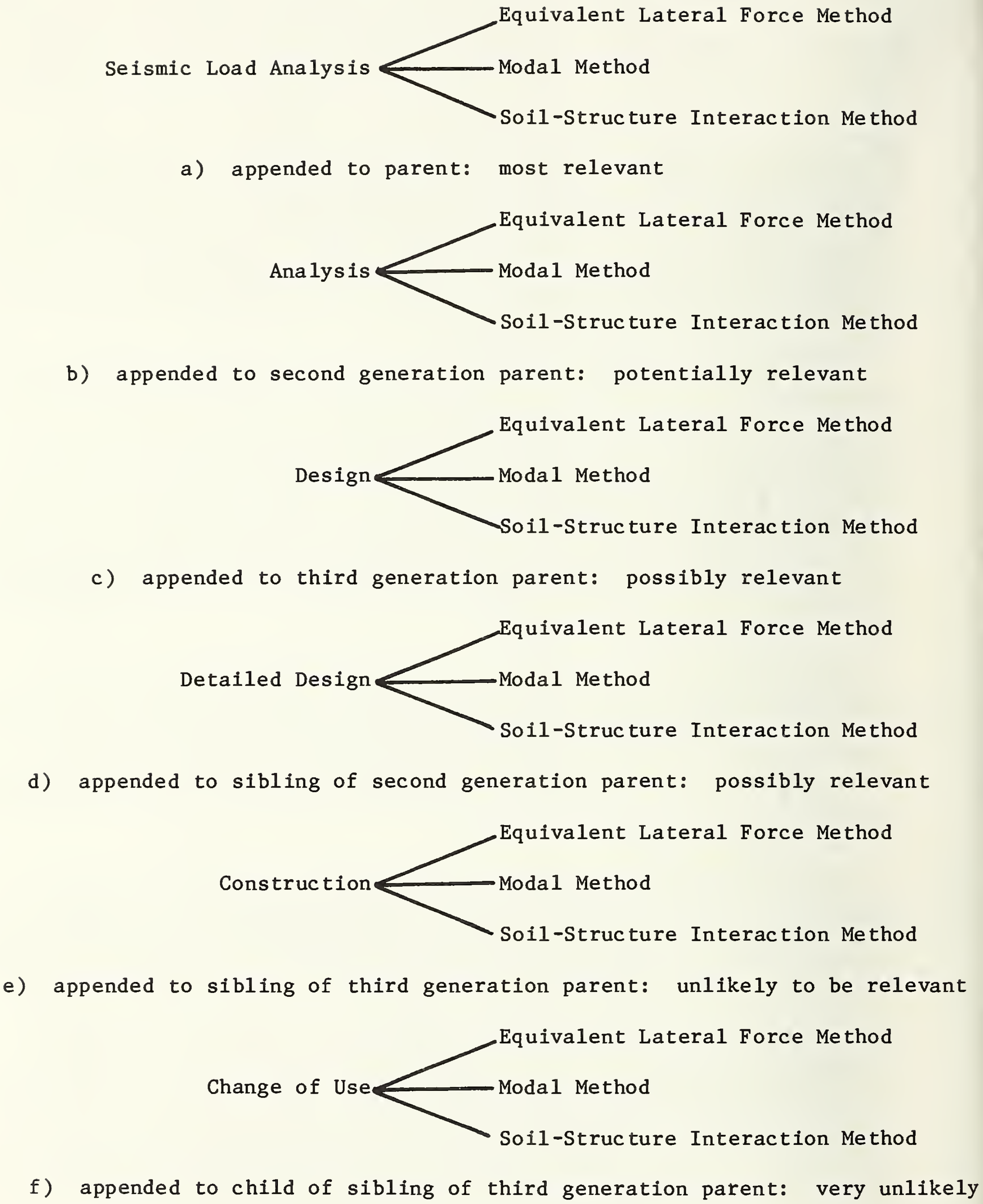
1) those whose parent is the root of a field
2) those whose parent is the root of a facet, but not a field
3) those whose parent is neither of the foregoing

Fields are independent classifications; there is no situation where appending the root nuclear tree of a field would be irrelevant, provided the logical rules are not violated. Facets that are not the root of a field are not independent; they are a subdivision of some other class or classes in their field. Thus it would be irrelevant to append the root of a facet unless such a class is already in the organizational tree. For example, "Cast-in-place" and "Precast" are the siblings in a nuclear tree that serves to modify the class "Concrete". Appending that nuclear tree to "Steel" would be irrelevant. As discussed in prior sections, the relations between facets within a field need further study. The problem addressed here, the eligibility of a facet to be appended into an organizational tree, appears to be simpler than the problem addressed in the previous section, the relevance of a classifier to an argument from a different facet within the same field.

A system for recording relations among facets is incorporated into the computer program described in the appendix. It is designed to meet the need to determine the eligibility of a facet. In that system, each facet (except the root of a field) is attached to one or more "foster parents," which are simply other classifiers in the field. The foster parents need not be terminal classifers on a facet. In addition, each link between a facet and a foster parent may be made conditional by specifying and recording "context" requirements. These requirements take the form of conditions that given classifiers must be present or absent (in the organizational tree) for the relation between the foster parent and facet to be relevant.

For example, the facet "Concrete Pile Construction" (shown in table 4.21) may be appended when its foster parent "Pile" is on the branch of the organizational tree only if the classifier "Concrete" is also on the branch. Such a system for checking the relevance between a classifier and an argument for the purpose of provision entry, the problem described in section 6.2.2, requires a more thorough specification of the permissible links for successful use.

For the nuclear tree that is not the root of a facet (the third type mentioned), the simplest rule for relevant appending is that the parent of the nuclear tree be present on the branch (not necessarily at the end) of the organizational tree. This has great intuitive appeal, but in practice it seems to be too rigid. Consider the classification for the field "Building Processes" used in the analysis of the seismic provisions [50] (see table 4.21b). Figure 6.2 shows several examples of appending the nuclear tree for the methods of "Seismic Load Analysis" to various classifiers in that field, with comments on the relative degree of relevance. The relevance depends a great deal on the specific context of the situation, but a few generalizations are in order. First, relevance is much less likely if what was defined as a cousin in the previous section is present on the branch. (Recall that a cousin


Seismic Load Analysis
Equivalent Lateral Force Method
Modal Method
Soil-Structure Interaction Method
Analysis
Equivalent Lateral Force Method
Modal Method
Soil-Structure Interaction Method
Design
Equivalent Lateral Force Method
Modal Method
Soil-Structure Interaction Method
Detailed Design
Equivalent Lateral Force Method
Modal Method
Soil-Structure Interaction Method
Construction
Equivalent Lateral Force Method
Modal Method
Soil-Structure Interaction Method
Change of Use
Equivalent Lateral Force Method
Modal Method
Soil-Structure Interaction Method

a) appended to parent: most relevant

b) appended to second generation parent: potentially relevant

c) appended to third generation parent: possibly relevant

d) appended to sibling of second generation parent: possibly relevant

e) appended to sibling of third generation parent: unlikely to be relevant

f) appended to child of sibling of third generation parent: very unlikely

Figure 6.2 Relevance in Appending Nuclear Trees

is a classifier from the same facet that is neither a predecessor or a descendent.) Second, relevance is possible even though the parent is not present, as long as some predecessor is present. Strict application of a rule requiring the parent to be present ignores this second observation, introducing unwanted depth in organizational trees and reducing desirable flexibility.

Before discussing techniques for generating organizational trees, it is useful to summarize the rules governing the use of nuclear trees. The following conditions should be satisfied when appending a nuclear tree to a node on an organizational tree:

1) Relevant: the root of a field may be appended at any location, the root of other facets may be appended when a class that the facet expands is present on the branch, and other nuclear trees may be appended when their predecessor is on the branch.

2) Unique: only one nuclear tree may be appended to any single node of an organizational tree.

3) Graded: a nuclear tree may not be appended where a descendant of the tree is already on the branch.

4) Complete: all the siblings shall be retained when appending the nuclear tree.

5) Progressive: the order of the siblings in the organizational tree shall be the same as in the original classification.

6) Minimal: larger nuclear trees should be appended after small nuclear trees.

It should be noted that the criteria for Relevant, Unique, and Complete frequently are not satisfied, and that different techniques for generating organizational networks are conveniently compared by the degree to which they satisfy the criteria.

Two criteria are available for making the decision to cease the extension of a branch and move on to the next branch: the absence of any qualified nuclear trees or the absence of an eligible provision. Nearly all techniques for generating outlines for an existing set of provisions make use of the second; they may be termed "provision driven." "Classifer driven" techniques allow one to maintain the completeness of the classification system in the organizational tree, and thus to check the completeness of the provisions. However, classifier driven techniques are not often used, except as described in chapter 5 for formulation of new standards, because the resulting organizational trees are extremely large and the majority of their branches do not have provisions associated with them.

The reason that classifier driven techniques develop so many empty branches is that the assumption of independence between the fields is not warranted. Consider the analysis of the seismic provisions [50], and the classification system for it given in table 4.21. If the 21 facets in the "Physical Entity" field were independent (obviously they are not, nor are they assumed so) an organizational tree fully incorporating all of them would have over three billion branches (the product of the terminal branches of the 21 facets). The most conservative estimate for the number of branches in an organizational tree for that classification would be as follows: assume that the "Physical Entity" field is one tree with no interaction (a false assumption), thus having 62 branches (the sum of the terminal classifiers of its facets), and that the "Process" field combines additively, not multiplicatively, with it to give all THINGS possible for subjects of Requirements (also a false assumption). Ignoring the "Limit State" and "Derived Value" fields, estimate the total number of branches as the product of the REQUIRED QUALITY field (12 branches) and the combined subject fields (79 branches). The total thus obtained is 948, whereas the total number of of Requirements found in that study in 242. Recall the discussion in section 4.1, that many of the requirements used in that study are of an improper, combined form. This means that there may be more than 242 basic requirements. It does not appear that the number would approach 948. The assumption regarding the "Physical Entity" field having only 62 branches is quite extreme. Thus, a more, realistic assumption would result in a number of branches far in excess of 948. On a smaller level, consider the study of the AISC Specification for Steel Design [97]. The classification, shown in table 3.3, has three fields with twelve, eight, and eight branches, with the product of 768 possible branches, assuming complete independence. Even assuming that two additively combine as subjects (once again, an extreme assumption) there are 160 possible branches to compare with the 42 criteria identified.

The assumption of independent fields is very convenient for constructing a classification. The job might well be too difficult without such as assumption. However, the assumption means that classifier driven techniques for generating organizational trees are not very useful, and thus, that systematic checks for completeness in the organizational trees are not profitable. There are many situations in which it is correct to append only a portion of a nuclear tree, leaving some of the siblings off the organizational network.

Provision driven techniques all depend, in some fashion, on the criteria used for entering provisions on a branch of an organizational tree. Past algorithms for automatically generating an organizational tree have all been provision driven. As stated in section 3.1.6, they used different techniques for tearing and appending trees and different criteria for entry of provisions. Some [96, 135] append entire fields of classifiers at a time, while another [97] appends trees one level at a time. Some [96, 97] require full relevance of all classifiers on a branch for the eligibility of a provision, while the other [135] does not require it, but frequently delivers it. Full relevance for entering classifiers is fairly easy to achieve with very simple classification systems. As mentioned previously, it is not so easy to achieve with complex classification schemes, and because complex schemes are necessary to clearly represent the scope of standards, the algorithms need to be re-examined.

None of the existing algorithms is strictly applicable to a faceted system in which only a subset of the facets will be included on any particular branch. They all assume that each facet (actually, field in the studies for which they were developed) will be appended on each branch and that the order of facets will be the same on each branch. Two algorithms [97, 135] produce branches without all facets in order to include all provisions in the outline. Neither will change the order of facets from a facet to a branch.

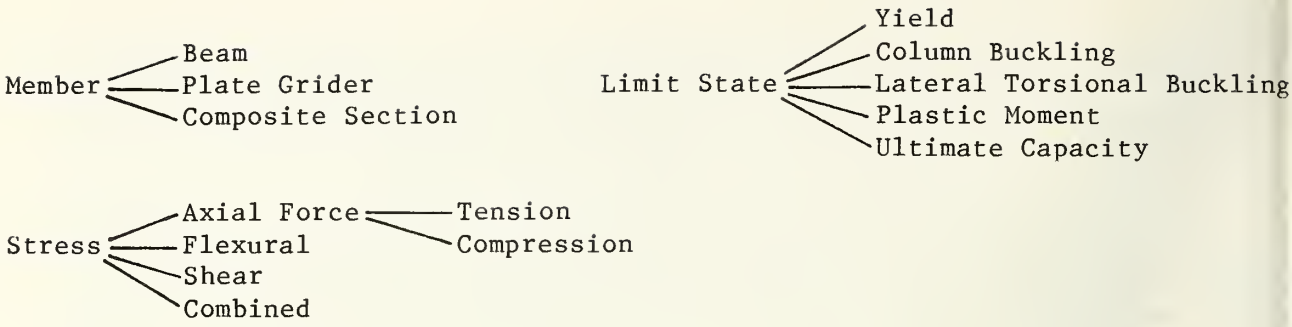
As shown in section 3.1.6 and figure 3.1, the algorithm that appends trees of classifiers one level at a time is able to generate branches that skip a level, if that is necessary to include a provision. Figure 6.3 shows another example based on the data and the technique presented in that report [97]. Note that the outline does not have the quality Unique. Placing "Yield," "Column Buckling," etc. as siblings of "Beam," "Plate Girder," and "Composite Section" introduces ambiquity into the organization. Given that classification system, the ambiguity is necessary in order to place provisions P1 through P4 in the outline, because the Graded criterion for provision entry (described in section 6.2.2) would rule out their entry in any outline in which the children of "Member" were at a higher level than the children of "Limit State."

The problem is that some provisions are of a more general nature than others, and therefore strict application of the Graded criterion for provision entry frequently will prevent the entry of the more general provisions in an organizational tree. It is possible to overcome this problem by expanding the classification system. Such was the reason for including the facet "Material Nature" with the sons "Material Generic" and "Material Specific" in the classification for the seismic provisions (table 4.21). This small facet was appended in conjunction with the material types thus:

```
Material Generic
Material Specific
          Wood
          Steel
          Concrete
          Masonry
```

This allowed otherwise identical branches to be constructed so that both general and specific provisions could be outlined without forcing "Wood," "Steel," etc. to become siblings of other classifiers to which they were logically unrelated.

The development of an algorithm to automatically and completely generate an organizational tree and outline in a provision driven mode that



a) abbreviated classifier trees

Provision	Arguments
P1	Member, Yield, Tension
P2	Member, Yield, Combined
P3	Member, Column Buckling, Compression
P4	Member, Plastic Moment, Combined
P5	Beam, Yield, Shear
P6	Beam, Lateral Torsional Buckling, Flexural

b) abbreviated argument list

```
Member
   Yield
      Axial Force
          Tension ......................P1
      Combined .........................P2
   Column Buckling
      Axial Force
          Compression ..................P3
   Plastic Moment
      Combined .........................P4
   Beam
      Yield
           Shear........................P5
      Lateral Torsional Buckling
           Flexural ....................P6
```

c) partial outline for the order: member, limit state, stress

Figure 6.3 Hypothetical Provision Driven Outline Illustrating Level Skipping.

could operate with a faceted classification system appears to be a formidable task. It has not been accomplished in this study, although no reason has been detected proving that such an algorithm is impossible. A semi-automatic, interactive computer algorithm has been developed. With respect to the criteria for appending nuclear trees presented earlier in this section, the algorithm operates as follows:

1) Relevant: no explicit check is made.

2) Unique: appending more than one nuclear tree to a single node is possible, but only upon explicit command of the operator.

3) Graded: no explicit check is made.

4) Complete: no explicit check is made in most situations -- this is not a classifier driven algorithm.

5) Progressive: the normal order of the classification is automatic, although it is possible for the operator to override it.

6) Minimal: the order of appending is completely up to the operator.

The lack of explicit checks for Relevant and Graded is not intended to imply that they are thought unnecessary. Rather, their incorporation is advocated. They are not in the prototype computer algorithm only because the algorithm was developed to test various criteria for provision entry rather than nuclear tree appending. A test for Relevant may be more problematical, as discussed. Various criteria should be tested for workability, because incorporation of such a test would be an important step in the development of a completely automatic algorithm.

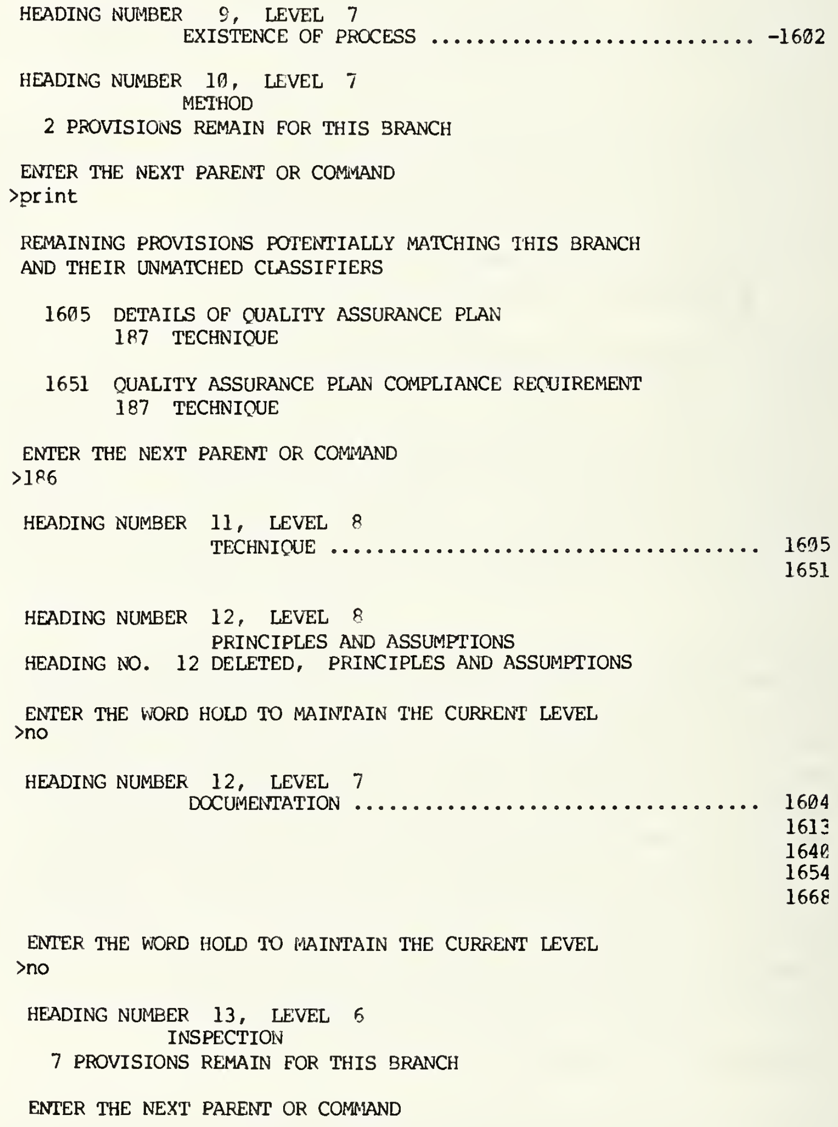
The interactive algorithm is provision driven. Once the operator has entered a branch, the algorithm determines which provisions have the potential to be outlined on the branch should additional classifiers be appended to it. The algorithm will also specify those classifiers upon request. The operator then continues by either appending another nuclear tree, or moving on to the next branch. Figure 6.4 shown a sample of the operator commands and computer responses involved in the generation of the outline shown in table 6.7. The algorithm is described in the appendix. It is designed to guide the application of the creative talents of an operator by using one step at a time. Significant improvements in the efficient use of an operator's time are possible, and the need for a completely automatic algorithm is not clear.

6.2.4 Comparison of Outlines

Once the organizational model generates several different outlines for the same set of provisions, comparisons are necessary. It is recognized that different users will be best suited by different qualities in

```
Classifier                                                    Provision

Building
     Required Qualities
          Building Processes
               Quality Assurance
                    Social Qualities . . . . . . . . . . . . . .    1601
                                                                    1625
                         Planning
                              Existance of Process . . . . . . .   -1602
```

a) outline prior to the sample runstream



```
HEADING NUMBER   9,  LEVEL  7
              EXISTENCE OF PROCESS ........................... -16Ø2

HEADING NUMBER  1Ø,  LEVEL  7
              METHOD
  2 PROVISIONS REMAIN FOR THIS BRANCH

ENTER THE NEXT PARENT OR COMMAND
>print

REMAINING PROVISIONS POTENTIALLY MATCHING THIS BRANCH
AND THEIR UNMATCHED CLASSIFIERS

  16Ø5  DETAILS OF QUALITY ASSURANCE PLAN
        187  TECHNIQUE

  1651  QUALITY ASSURANCE PLAN COMPLIANCE REQUIREMENT
        187  TECHNIQUE

ENTER THE NEXT PARENT OR COMMAND
>186

HEADING NUMBER  11,  LEVEL  8
                TECHNIQUE .................................... 16Ø5
                                                               1651

HEADING NUMBER  12,  LEVEL  8
                PRINCIPLES AND ASSUMPTIONS
HEADING NO.  12 DELETED,  PRINCIPLES AND ASSUMPTIONS

ENTER THE WORD HOLD TO MAINTAIN THE CURRENT LEVEL
>no

HEADING NUMBER  12,  LEVEL  7
              DOCUMENTATION .................................. 16Ø4
                                                               1613
                                                               164Ø
                                                               1654
                                                               1668

ENTER THE WORD HOLD TO MAINTAIN THE CURRENT LEVEL
>no

HEADING NUMBER  13,  LEVEL  6
            INSPECTION
  7 PROVISIONS REMAIN FOR THIS BRANCH

ENTER THE NEXT PARENT OR COMMAND
```

b) sample command and response runstream

Figure 6.4 Example of Interactive Construction of an Outline (The example shows a portion of the outline of table 6.7)

an outline, and not much guidance is offered here for classifying the differences in users. It is possible to discuss various criteria for goodness of outlines in terms of the objectives for an organization. The relative weight of the various objectives generally would depend on the user.

The first two objectives, Relevant and Meaningful, are best discussed together. The first, Relevance, is a key factor in generating outlines, and both are of the utmost importance in developing a classification. Also, they are important selectors between outlines in that certain users may prefer outlines because "more relevant" classes are used for the major subdivisions of the outline. For design standards, for example, practitioners of structural design usually prefer major subdivisions based on Physical Entities while professors instructing students in the theory behind standard practice frequently prefer major subdivisions based on performance qualities, such as Limit States. Such opinions are frequently quite strong, but no adequate measure or strategy for these qualities can now be offered. Care must be taken that they are considered.

The logical objectives Unique and Complete offer easily measurable comparisons. Outlines with considerable repetition of provisions are more ambiguous, and therefore less unique, than outlines without repetition. Outlines that fail to provide headings for all provisions are obviously less complete than those that do. Beyond these simple tests, outlines can also be compared for uniqueness by examing the frequency of "step-sibling" headings. (Recall that this is introduced when more than one nuclear tree is appended to a single node in the organizational network.) Although it is more a measure of completeness of the provisions than the outline, outlines can be tested for completeness by examing the frequency of incomplete sets of sibling headings.

Graded is an objective that offers little opportunity for comparison. It is a factor in preparation of a classification and in generating outlines. Progressive is primarily a factor in preparation of a classification, but not entirely. It may be possible to generate a measure of the distortion introduced in the cross references of the information network by the arrangement given by an outline, as discussed in section 6.3. Such a measure might be useful as a measure of the quality Progressive.

Outlines may be compared for the quality Intelligible by examing their depth and breadth. However the comparison should not be carried out prior to condensing unneeded levels out of the outline, as described in sections 5.3 and 6.2.1. Recall that "seven plus or minus two" was offered as a limit on the number of siblings at a level in section 4.2. It also appears to be an appropriate limit on the number of levels [87, 139]. Experience indicates that the number of levels in an organizational tree often exceeds this limit, but that extensive condensation is usually possible. Table 6.9 contains such a condensation taken from the study of the seismic provisions [50]. Note that the number of sibling headings in an organizational tree can be increased over that in the classification by appending more than one nuclear tree to a single node, and also note that this number may be increased again by the process

Table 6.9 Condensation from Organizational Network to Final Outline

Classifiers	Outline	Provisions
Building		
Required Qualities	1.1 General Performance Requirements for Buildings	Section 1.1
Building Processes		
Regulation	1.2 Regulatory Procedures and Parameters	
Technique	1.2.1 Acceptance of Alternates	1510
Regulatory Parameters		
Scope	1.2.2 Scope	-1210
Ground Motion	1.2.3 Ground Motion	-1405
		-1415
		-1425
Classification	1.2.4 Hazard Classification	-1430
		-1490
Development and Use	1.3 General Requirements for the Development and Use of Buildings	1305
Development		
Design		
Method		
Technique	1.3.1 Load Combination	1315
Documentation	1.3.2 Design Documentation	Part of 1305
Specific Buildings		
Proposed (New)	1.4 Development of New Buildings	1345
Design	1.5 Design of New Buildings	
System		
Structural	1.5.1 Structural Design	1365
Non-structural	1.5.2 Non-structural Design	8001
Material	1.5.3 Materials of Construction	1370
Construction	1.6 Construction of New Buildings	
Quality Assurance	1.7 Quality Assurance for New Buildings	
Social Qualities		1601
		1625
Existence of Process	1.7.1 Procedures Required	
Planning (QA)		1602
Inspection		
Testing		1637

Table 6.9 continued

Method			
Technique	1.7.2	Techniques of Quality Assurance	1651
Planning (QA)			1605
Inspection			-1628
Testing			-1635
			-1641
			-1644
Documentation	1.7.3	Documentation of Quality Assurance	1640
			1654
			1668
Planning (QA)			1604
			1613
Inspection			1655
			1662
Testing			1674
Use			
Specific Buildings			
Existing	1.8	Use of Existing Buildings	
Alteration	1.8.1	Alteration and Repair	
Strength Required			1380
Repair			
Change of Use	1.8.2	Change of Use	
Strength Required			1390
Hazard Abatement	1.8.3	Systematic Hazard Abatement	13001
Specific Buildings	1.9	Special Performance Requirements for Specific Buildings	
Group III			
Dysfunction of DSS	1.9.1	Group III Functional Requirement	1469
Access/Egress Blocked	1.9.2	Group III Access Requirement	1472
Category D			
Ground Rupture	1.9.3	Category D Site Limitation	1493

of condensation. Intelligible is the only objective that requires condensation of an organizational tree.

The quality Minimal is easily tested by comparing the number of headings in outlines, after condensation. The quality Even is quite subjective, although the number of headings per chapter, the range of subdivision, and similar issues can be examined. It seems that the structure of the classification system is the place where it comes into play more than in comparing outlines. However, outlines can and will be criticized as being less even than others. The comparison should be on the condensed outline, if condensation is used at all.

Examples comparing outlines prepared with different entry criteria for provisions are offered in tables 6.7 and 6.8 (see section 6.2.2). More often, comparisons will be between outlines with different ordering of the classifiers on the branches. Tables 6.5 and 6.7 offer such a comparison. Some comparisons may be made on the effect of dropping certain classifiers from the classification system, as was done with the classifier "Scope" in table 6.5, 6.7, and 6.8.

In some instances selected classifiers may be alternately dropped and reinstated for the purpose of "fine tuning" an outline. Similarly, the criteria for provision entry may be changed back and forth. Such juggling is usually done to include all provisions exactly one time in a given outline. In such instances, it would be prudent to have some indication printed on the outline where the status of classifiers or entry criteria is being changed. The computer program described in the appendix does not print such information.

6.3 Organization with the Information Network

All the terminal nodes of an information network must be Requirements: by definition, the status of compliance with a standard, or a portion of a standard, is just what the terminal nodes are. Not all requirements need to be terminal nodes of an information network, however. As found in section 4.1.3, synthetic Requirements are equivalent to a new basic Requirement, meaning that a Requirement may be an ingredient of another Requirement. Furthermore, there is no reason that a Requirement could not be an ingredient of a Determination. In order to guarantee access to all provisions, it is necessary for every terminal node in the information network to be included in the organizational system. Generally speaking, it is relevant for all Requirements to be included in the organizational system, and it may be useful to include some or all of the Determinations.

If only the terminal nodes are in the organizational system, there is no overlap between the information network and the organizational system. The expression of the organization may proceed in two steps: the terminal nodes are grouped and ordered by the outline method, and the remainder of the provisions are ordered according to the global ingredience networks for each of the terminal nodes. (Recall that two strategies, conditional and direct, exist for ordering provisions in a global ingredience network, as discussed in section 2.3; also see figures 2.6 through 2.9.)

Frequently the organizational system does include more than the terminal nodes. In those situations, the grouping and ordering of the outline takes priority over that of the information network, and the information network provides the necessary cross references. The analysis of the seismic provisions [50] provides a good example of the interaction between the two systems. Figure 6.5 is a computer printed information network for a portion of the provisions. Each node shown in the network is a Determination; input nodes are not shown in this printing, and there are no Requirements in that portion of the seismic provisions. The datum number has a special relation to the original organization of the seismic provisions:

1) the first digit represents the chapter number; each of the nodes in figure 6.5 is from chapter 4.

2) the second digit represents the section within the chapter; thus, 4205 is from section 4.2.

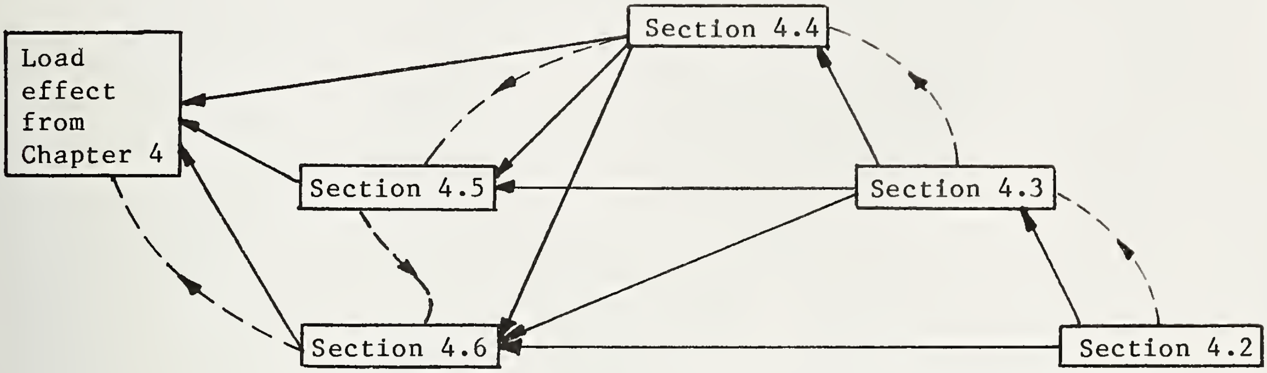
3) the third and fourth digits are arbitrarily assigned, but they do represent a progressive sequence; thus, 4208 is located between 4205 and 4210.

Figure 6.6 shows that the original ordering of the provisions follows two trends that are based on the information network. Figure 6.6a shows that the ordering of sections is equivalent to the direct order, defined in figure 2.8. Figure 6.6b shows that the detailed level ordering within section 4.2 of the seismic provisions is nearly equivalent to conditional order, defined in figure 2.7. The detailed level ordering within the other sections is also equivalent to conditional order. Such a result could be obtained in a systematic fashion by including the root Determination for each of the sections in the outline system and then ordering the remainder of the nodes according to the conditional strategy. Direct order among the sections would not necessarily be the outcome of applying the outline system, but it could be. The outline system allows one to override the natural order of the information network in several ways.

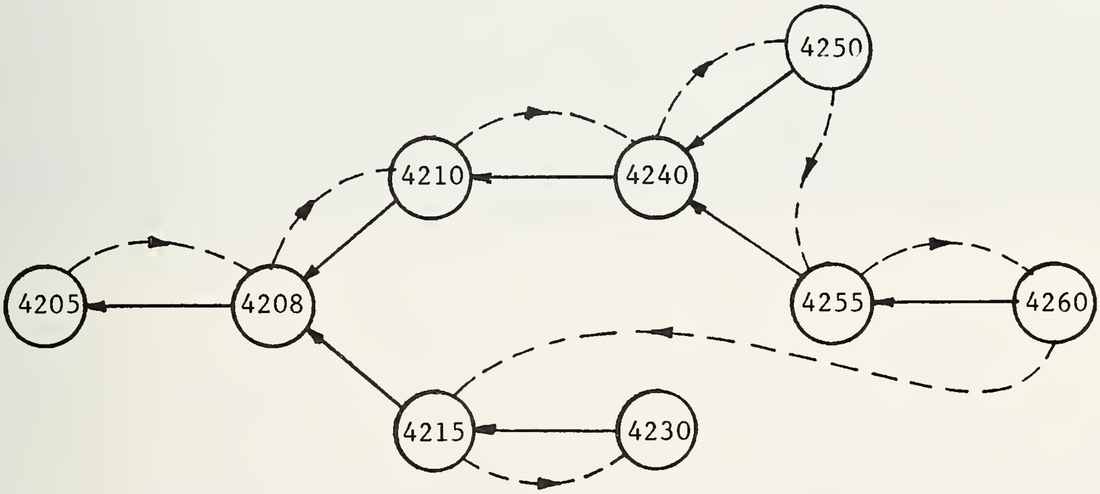
As more nodes with dependents are included in the outline, more distortion of the information network is possible. It does not appear wise to overlook the natural order of the information network when comparing outlines, because distortion is likely to increase the number of cross references necessary. A simple measure of the cross references introduced by an outline has been defined thus: consider each provision in the outline to occupy one position in a sequence, then sum the number of positions between each ingredient and dependent, ignoring those ingredients not included in the outline. Unfortunately, the measure does not appear to be a sensitive indicator of a desirable progression of provisions. More study of the subject is desirable, because such a measure is likely to exist.

```
GLOBAL INGREDIENTS OF COMPLETE NETWORK

EXTREME LEVEL FROM OUTPUT

   Ø   1   2   3   4   5   6   7   8   9   1Ø  11  12  13  14  15

   4ØØ1  EQUIVALENT LATERAL FORCE ANALYSIS REQUIREMENT
   :...456Ø  OVERTURNING MOMENT REQUIREMENT
   4Ø1Ø  EARTHQUAKE LOAD EFFECT FROM ELF/MODAL ANALYSIS
   :.......442Ø  STORY SHEAR FORCE EFFECT
   :       :.......441Ø  SEISMIC STORY SHEAR
   :               :...............431Ø  SEISMIC STORY FORCE
   :                               :...42Ø5  SEISMIC BASE SHEAR
   :                               :   :...42Ø8  ELF SEISMIC BASE SHEAR WITHOUT SOIL STRUCTURE INTERACTION
   :                               :       :...421Ø  SEISMIC DESIGN COEFFICIENT
   :                               :       :   :...424Ø  BUILDING PERIOD
   :                               :       :       :...425Ø  CALCULATED FUNDAMENTAL BUILDING PERIOD
   :                               :       :       :...4255  APPROXIMATE BUILDING PERIOD
   :                               :       :           :...426Ø  COEFFICIENT FOR APPROXIMATE PERIOD
   :                               :       :...4215  TOTAL GRAVITY WEIGHT OF BUILDING
   :                               :           :...423Ø  EFFECTIVE SNOW LOAD
   :                               :...432Ø  VERTICAL DISTRIBUTION FACTOR
   :                                   :...434Ø  TOTAL WEIGHT AT LEVEL X
   :                                   :   :..-4215* TOTAL GRAVITY WEIGHT OF BUILDING
   :                                   :...433Ø  VERTICAL DISTRIBUTION EXPONENT
   :                                       :......-424Ø* BUILDING PERIOD
   :...445Ø  TORSIONAL MOMENT EFFECT
   :   :...446Ø  TORSIONAL MOMENT
   :       :......-441Ø* SEISMIC STORY SHEAR
   :       :...448Ø  ACCIDENTAL TORSIONAL MOMENT
   :           :..-441Ø* SEISMIC STORY SHEAR
   :...451Ø  OVERTURNING MOMENT EFFECT
   :   :...4515  OVERTURNING MOMENT AT LEVEL X
   :   :   :...452Ø  ELF OVERTURNING MOMENT AT LEVEL X
   :   :       :...453Ø  OVERTURNING MOMENT REDUCTION FACTOR
   :   :       :..................-431Ø* SEISMIC STORY FORCE
   :   :..........-441Ø* SEISMIC STORY SHEAR
   :   :..-442Ø* STORY SHEAR FORCE EFFECT
   :...........464Ø  STABILITY COEFFICIENT
   :           :...4645  TOTAL GRAVITY LOAD ABOVE LEVEL X
   :           :   :............................-4215* TOTAL GRAVITY WEIGHT OF BUILDING
   :           :...46Ø5  FIRST ORDER DESIGN STORY DRIFT
   :           :   :...461Ø  DEFLECTION AT STORY X
   :           :       :...46Ø8  ELF DEFLECTIONS WITHOUT SOIL STRUCTURE INTERACTION
   :           :           :...4615  ELASTIC DEFLECTION AT STORY X
   :           :               :......................-425Ø  CALCULATED FUNDAMENTAL BUILDING PERIOD
   :           :               :......................-4255* APPROXIMATE BUILDING PERIOD
   :           :               :..-431Ø* SEISMIC STORY FORCE
   :           :               :...463Ø  REDUCED SEISMIC FORCES CORRESPONDING TO CALCULATED PERIODS
   :           :                   :..................-425Ø  CALCULATED FUNDAMENTAL BUILDING PERIOD
   :           :                   :..-42Ø5* SEISMIC BASE SHEAR
   :           :                   :..-432Ø* VERTICAL DISTRIBUTION FACTOR
   :           :..-441Ø* SEISMIC STORY SHEAR
   :...4665  INCREASE IN FORCE EFFECTS FROM SECOND ORDER EFFECTS
       :...466Ø  DESIGN STORY DRIFT
           :..-464Ø* STABILITY COEFFICIENT
           :......-46Ø5* FIRST ORDER DESIGN STORY DRIFT
           :...465Ø  INCREMENTAL FACTOR FOR SECOND ORDER EFFECTS
   4522  OVERTURNING MOMENT AT FOUNDATION WITHOUT REDUCTION
   :..............-453Ø  OVERTURNING MOMENT REDUCTION FACTOR
   :..............................-431Ø* SEISMIC STORY FORCE
```

Figure 6.5 Information Network for Equivalent Lateral Force Analysis



a) simplified information network showing direct ordering path as dotted line



b) information network for section 4.2 showing conditional ordering as dotted line

Figure 6.6 Example Showing Direct Ordering Imposed on an Information Network by the Organization.

Facing page: The wide variety of buildings to which standards must apply requires sound principles for organizing standards.

CHAPTER 7

CONCLUSIONS

The organization of a standard deals with both the scope and the arrangement of the provisions it includes. In terms of scope, the organization helps define the intended qualities for the intended subjects. In terms of arrangement, the organization helps the user find with confidence the relevant provisions.

Organization has objective qualities that allow it to be treated formally. The decision table and information network systems for analysis and representation of the provisions of a standard provide the context and need for an organizational system that capitalizes on these objective qualities. Existing models of organizational systems offer opportunities for improvement. The development of an improved model is aided by the sciences of classification and linguistics, and is based principally on the following findings:

1) Five necessary and four desirable qualities for an organization are identified, verified, and adopted as objectives and guidelines for organization. The qualities pertain specifically to the headings used to provide access to provisions; they are Relevant, Meaningful, Unique, Complete, Graded, Progressive, Intelligible, Minimal, and Even.

2) Provisions in a standard can be grouped into two functional groups, Requirements and Determinations. Requirements can be further grouped, into Basic, Multiple, Cumulative, Application, Synthetic, and Mixed Requirements based on their structure and interrelation with other Requirements. This grouping makes possible a clear definition of the interface of the organizational system with the decision table and information network systems and provides some of the basis for systematic classification. Further, it provides the basis for recommending a change in the manner of representing provisions with decision tables, to wit: only Basic Requirements, Synthetic Requirements (which are equivalent to Basic Requirements), or Application Requirements that are too complex for the organizational system should be used.

3) A relevant basis is found for classifying Requirements based on an idealized model of the relation between syntax and semantics, which shows that a Basic Requirement names a THING as its subject and contains a REQUIRED QUALITY for that subject as its predicate. The idealized model appears to be the key step in providing reproducibility in the organizational system. A secondary benefit derived from the linguistic analysis performed in support of this model is the added insight on the degree of exhaustiveness appropriate to the formal representation of standards.

4) A structure is recommended for the classification system that meets the conflicting demands of users with dissimilar purposes or backgrounds. The faceted structure provides a clear division, which is necessary for systematic use, between levels that are strictly logical and those that are not. The major divisions of the structure (the fields) are easily correlated with the model structure for Requirements. The system promotes several of the qualities identified as objectives, has a background in highly developed areas of classification science, and is flexible enough to accomodate special situations and uses.

5) Basic categories for classification are derived from the model of a Requirement, the functional roles of provisions, and philosophical considerations. They are expressed for

engineering design standards for buildings to allow a meaningful starting point for organization.

6) A technique for systematic formulation of new standards is developed. The two essential steps are (1) top down classification as the scope is defined and (2) construction of an organizational tree from the classification for the purpose of identifying potential requirements. The technique is based on the performance concept, but may be used in the development of standards that are not performed oriented.

7) Procedures are developed for forming an index as one expression for the organization. An index is most useful if all types of provisions are included. It is found that some rules for classing provisions for indexing differ from those for other aspects of organization.

8) Procedures are developed for forming outlines as another expression of the organization, and appropriate measures are defined for the comparison of alternate outlines for the same standard. Criteria for placement of provisions in outlines and for construction of outlines from the classification are proposed to promote and preserve the objectives of organization. The identification of the nuclear tree as the smallest logical unit of a classification introduces rigor into the techniques for tearing and appending trees of classifiers used to form an outline. The rudiments of a system for recording relations between facets also are identified. Various outlining techniques are explored, and a computer algorithm for an interactive style of operation is developed and tested. It is found that the independence assumed between fields in a classification system is not rigorous enough to support systematic analyses of an existing set of provisions for completeness based on the scope of the classification system.

Several areas worthy of further study are identified. A brief summary of these includes:

1) The model for the structure of a Basic Requirement and the basic categories have been developed almost entirely on the basis of studies of provisions from building standards, mostly from the domain of structural engineering. The applicability and extension of these concepts to other types of standards needs careful attention.

2) A structure for rigorously defining relations between facets is needed, particularly for efficient checking of relevance when considering a provision for entry on a branch of an organizational tree.

3) An algorithm for the fully automatic generation of an outline appears possible and useful.

4) A measure of the distortion in cross referencing created by a particular outline should be developed.

5) The strategy for generating multiple level indexes should be formalized and tested.

Other areas could be studied, such as searching for a model structure for Determinations (or finding a better name for such provisions, for that matter) or exploring the typical deviations from logical classification, but they all are of a lower priority. A higher priority should be given to further testing of the recommendations arising from this study. Many of them were not clearly formulated until the last major case study was complete.

Acknowledgements

This report is nearly identical to the dissertation prepared by J. R. Harris under the supervision of R. N. Wright and submitted in May, 1980, to the Graduate College of the University of Illinois at Urbana-Champaign in partial fulfillment of the requirements for the degree Doctor of Philosophy in Civil Engineering. The National Science Foundation contributed support to this research through grant number PFR-14698 for the analysis of the tentative seismic provisions.

The authors extend their appreciation to Dr. John W. Melin, Professor of Civil Engineering at the University of Illinois, and Dr. Steven J. Fenves, University Professor of Civil Engineering at Carnegie-Mellon University, for their many valuable suggestions. Dr. Kirk Rankin, formerly of the National Bureau of Standards, provided guidance in the field of linguistics, which is gratefully acknowledged.

REFERENCES

1. A Classification Framework for the Construction Industry, Directorate General of Development, Department of the Environment (U.K.), London, 1972.

2. A Decent Home, The Presidents Commission on Urban Housing (the Kaiser Commission), Washington, December, 1968.

3. ACI Committee 118, "Decision Logic Table Format for Building Code Requirements for Reinforced Concrete (ACI 318-71)," ACI Journal, Vol. 70, No. 12, December, 1973, pp. 788-792.

4. Alexander, C., Notes on the Synthesis of Form, Harvard University Press, 1964.

5. Allen, D.E., "Limit State Design - A Unified Procedure for the Design of Structures," Engineering Journal, Engineering Institute of Canada, February, 1970.

6. Allen, D.E., "Limit States Design - A Probablistic Study," Canadian Journal of Civil Engineering, Vol. 2, 1975, page 36.

7. Applied Technology Council, Tentative Provisions for the Development of Seismic Regulations for Buildings, (ATC-3-06), National Bureau of Standards, Special Publication 510, 1977.

8. Ayers, C., SPECIFICATIONS: for Architecture, Engineering, and Construction, McGraw-Hill, New York, 1975.

9. Bach, Emmon, An Introduction to Transformational Grammars, Holt, Rinehart, and Winston, New York, 1964.

10. Bar-Hillel, Yehoshua, Language and Information, Addison-Wesley, Reading, Mass., 1964.

11. Basic Building Code (1978 edition), Building Officials and Code Administrators International, Chicago, Illinois, 1978.

12. Beineke, L.J., and R.J. Wilson, Selected Topics in Graph Theory, Academic Press, New York, 1978, p.10.

13. Bliss, H.E., The Organization of Knowledge, Henry Holt & Company, New York, 1929.

14. Brill, M., "Techniques for Developing Performance Specifications for Buildings," Performance Concept in Buildings, National Bureau of Standards, Special Publication 361, Vol. 1, 1972, pp. 171-180.

15. Building Code Requirements for Design Loads on Buildings and Other Structures, (ANSI A58.1-1972), American National Standards Institute, New York, 1972.

16. Building Code Requirements for Reinforced Concrete (ACI 318-71), American Concrete Institute, Detroit, Michigan, 1971.

17. Building Codes: A Program for Intergovernmental Reform, Advisory Commission on Intergovernmental Relations, Washington, January, 1966.

18. Building the American City, U.S. National Commission on Urban Problems (Douglas Commission), Washington, December, 1968.

19. Carnap, Rudolph, Philosophical Foundations of Physics, Basic Books, New York, 1966.

20. Carter, D.J., and B. Whitehead, "A Study of Pedestrian Movement in a Multi-Storey Office Building," Building and Environment, Vol. 11, pp. 239-247, Pergamon Press, Great Britain, 1976.

21. Chomsky, Noam, Syntactic Structures, Mouton & Co. 'S-Gravenhage, The Hague, The Netherlands, 1957.

22. Chomsky, Noam, Aspects of the Theory of Syntax, MIT Press, Cambridge, Massachusetts, 1965.

23. Chomsky, Noam, Cartesian Linguistics, Harper & Row, New York, 1966.

24. Cooke, P.W., and R.M. Eisenhard, A Preliminary Examination of Building Regulation Adopted by the States and Major Cities, National Bureau of Standards, NBSIR 77-1305, 1977.

25. Cunningham, L.K., J.W. Melin, R.L. Tavis, "Detailed Application of a Technology for the Formulation and Expression of Standards," Civil Engineering Studies, No. SRS 446, University of Illinois, Urbana, Illinois, 1978.

26. Deese, James, "Thoughts into Speech," American Scientist, Vol. 66, May-June, 1978, pp. 314-321.

27. "Development of a Classification System for the Building Industry," NBSIR Information Sheet, National Building Research Institute, (South Africa), Pretoria, 1975.

28. Driedger, K.C., "Memorandum and the Drafting of Acts of Parliament and Subordinate Legislation," Department of Justice (Canada), Ottowa, Canada, 1951.

29. Dunham, C.W., and R.D. Young, Contracts, Specifications, and Law for Engineers, McGraw-Hill, New York, 1971 (2nd ed.).

30. Encyclopedia Britannica, 15th Edition, 1974.

31. Falk, D., "Building Codes in a Nutshell," Real Estate Review, Fall, 1975.

32. Federal Register, "Occupational Safety and Health Standards: Means of Egress; Hazardous Materials; and Fire Protection," Vol, 43, No. 247, December 22, 1978.

33. Fenves, S.J., "Tabular Decision Logic for Structural Design," Journal of the Structural Division, ASCE, Vol. 92, No. ST6, Proc. Paper 5024, December 1966, pp. 473-490. Also in Fenves, Computer Methods in Civil Engineering, Prentice Hall, Englewood Clifts, N.J., 1967.

34. Fenves, S.J., "Performance Requirements for Standards Processing Software," Report R-79-111, Department of Civil Engineering, Carnegie-Mellon University, Pittsburgh, Pennsylvania, 1979.

35. Fenves, S.J., "Functional Specifications for Standards Processing Software," Report R-120-679 SJF, Department of Civil Engineering, Carnegie-Mellon University, Pittsburgh, Pennsylvania, 1979.

36. Fenves, S.J., E.H. Gaylord and S.K. Goel, "Decision Table Formulation of the 1969 AISC Specification," Civil Engineering Studies, No. SRS 347, University of Illinois, 1969.

37. Fenves, S.J., and Thanet Norabhoompipat, "Potentials for Artificial Intelligence Applications in Structural Engineering Design and Detailing", in Artificial Intelligence and Pattern Recognition in Computer-Aided Design, J.C. Latombe, ed., North Holland Publishing Co., 1978, pp. 105-121.

38. Fenves, S.J., K. Rankin, H.K. Tejuja, The Structure of Building Specifications, Building Science Series 90, National Bureau of Standards, Washington, D.C., September, 1976.

39. Fenves, S.J., and R.N. Wright, "The Representation and Use of Design Specifications," Symposium on Structural and Geotechnical Mechanics, University of Illinois, Urbana, Illinois, October 2-3, 1975, Also published as National Bureau of Standards Technical Note 940, Washington, D.C., June, 1977.

40. Ferguson, R.S., and C.C. Gordon, "Catalog of Building Safety Instruments," Research and Innovation in the Building Regulatory Process, Proceedings of the First NBS/NCSBCS Joint Conference held in Providence, RI, Sept. 1976, P.W. Cooke, ed., National Bureau of Standards Special Publication 473, 1977, pp. 259-278.

41. Field, C.G., and S.R. Rivkin, The Building Code Burden, D.C. Heath and Co., Lexington, Mass., 1975.

42. Field, C.G., and F.T. Ventre, "Local Regulation of Building: Agencies, Codes, and Politics," 1971 Municipal Year Book, International City Management Association, Washington, 1971, pp. 139-165.

43. Galambos, T.V., "Proposed Criteria for Load and Resistance Factor Design of Steel Building Structures," Research Report No. 45, Civil Engineering Department, Washington University, St. Louis, Missouri, 1976.

44. Gardner, Martin, "On Charles Sanders Peirce: Philosopher and Gamesman," Scientific American, Vol. 234, No. 1, July 1977, pp. 18-26.

45. Goel, S.K., and S.J. Fenves, "Computer Aided Processing of Structural Design Specifications," Civil Engineering Studies, No. SRS 348, University of Illinois, 1969.

46. Goel, S.K., and S.J. Fenves, "Computer-Aided Processing of Design Specifications," Journal of the Structural Division, ASCE, Vol. 97, No. ST1, Proc. Paper 7839, January 1971., pp. 463-480.

47. Goel, S.K., S.J. Fenves, and E.H. Gaylord, "Adapting the AISC Specification to Computer Aided Design," Engineering Journal, American Institute of Steel Construction, Vol. 8, No. 3, July Third Quarter, pp. 80-89, 1971.

48. Harper, G.N., "AUTOSPEC: Automated Preparation of Specifications," Journal of the Structural Division, ASCE, Vol. 92, No. ST6, Proc. Paper 5002, December 1966, pp. 45-53.

49. Harris, J.R., "Logical Analysis of Building Code Provisions," in Research and Innovation in the Building Regulatory Process, Proceedings of the First NBS/NCSBCS Joint Conference held in Providence, RI, Sept. 1976, P.W. Cooke, Ed., National Bureau of Standards Special Publication 473, 1977, pp. 285-316.

50. Harris, J.R., S.J. Fenves, R.N. Wright, Analysis of Tentative Seismic Design Provisions for Buildings, National Bureau of Standards Technical Note 1100, 1979.

51. Harris, J.R., J.W. Melin, R.L. Tavis, R.N. Wright, "Technology for the Formulation and Expression of Specifications, Volume I: Final Report," Civil Engineering Studies, No. SRS 423, University of Illinois, Urbana, Illinois, 1975.

52. Harris, J.R., J.W. Melin, and C. Albarran, "Technology for the Formulation and Expression of Specifications, Volume II: Program User's Manual," Civil Engineering Studies, No. SRS 424, University of Illinois, Urbana, Illinois, 1975.

53. Harris, J.R., and R.N. Wright, "Systematic Organization of Standards and Codes," in Research and Innovation in the Building Regulatory Process, Proceedings of the Second NBS/NCSBCS Joint Conference held in Bozeman, Montana, Sept. 1977, P.W. Cooke, Ed., National Bureau of Standards Special Publication 518, 1978, pp. 145-160.

54. Harter, Phillip, "Voluntary Standards Used in Regulation," ASTM Standardization News, Vol. 5, No. 5, May, 1977.

55. Hattis, David B., and Thomas E. Ware, The PBS Performance Specification for Office Buildings, National Bureau of Standards Report 10 527, September 1971.

56. Hempel, Carl Gustav, "Fundamentals of Concept Formulation in Empirical Sciences," International Encyclopedia of Unified Science, Vol. 2, No. 7, University of Chicago Press, Chicago, 1952.

57. Hempel, Carl Gustav, Aspects of Scientific Explanation, Free Press, New York, 1965.

58. Hubble, E.P., The Realm of the Nebulae, Dover Publications, New York, 1958.

59. Interim Performance Criteria for Solar Heating and Combined Heating/Cooling Systems and Dwellings, National Bureau of Standards, Washington, D.C. 1975.

60. Irvine, R.R., "The Constitutional and Legal problems Surrounding the Use of National Codes and Standards by States and Municipalities," State Laws and Ordinances, American Standards Association, New York, 1949, pp. 31-35.

61. Jevons, W.S., The Principles of Science, MacMillan and Co., New York, 1874.

62. Kant, Immanuel, Prolegomena to Any Future Metaphysics (originally published 1783), translated by L.W. Beck, Boss-Merril Company, Indianapolis, 1950.

63. Kapsch, R.J., unpublished memorandum, National Bureau of Standards, Washington, D.C., December 18, 1974, (to R.D. Dikkers, Re: Comments on the Interim Performance Criteria for the Design and Evaluation of Solar Heating and Cooling Systems and Dwellings).

64. Kapsch, Robert J., Existing Architectural Information Indexing Systems, National Bureau of Standards, NBSIR 76-1064, 1976.

65. Kapsch, R.J., "Feasibility of the Use of Numerical Taxonomy in the Study of Buildings," unpublished M.S. thesis, George Washington University, 1977.

66. Karlen, I., and K.L. de Vries, "International Work on Building Information: Present and Future (W52)," in Construction Research International, Proceedings of the Seventh CIB Triennial Congress, Edinburgh, Scotland, Sept., 1977, Vol. 1, Keith Alsop, ed., The Construction Press Ltd., Lancaster, England.

67. Knuth, D.E., The Art of Computer Programming, Vol. 1, Fundamental Algorithms, 2nd edition, Addison-Wesley, Reading, Mass. 1973.

68. Korner, Experience and Theory, Routledge and Kegan Paul Ltd., London, 1966.

69. Korner, Stephen, "Classification Theory," in Encyclopedia Britannica, 15th Edtion, Volume 4, page 691, 1974.

70. Kornsand, N.J., "A Consulting Engineer's View of the Building Code Process from Conception to Adoption", in Research and Innovation in the Building Regulatory Process, Proceedings of the Second NBS/NCSBCS Joint Conference held in Bozeman, MT, Sept., 1977, P.W. Cooke, ed., National Bureau of Standards Special Publication 518, 1977, pp. 67-76.

71. Koutsoudas, Andreas, Writing Transformational Grammar, McGraw-Hill, New York, 1966.

72. Lane, N.D., An Evaluation of Architectural Information Systems, Army Construction Engineering Research Laboratory, Champaign, Illinois, 1974.

73. Langacker, Ronald W., Language and Its Structure, Harcourt, Brace, and World, New York, 1968.

74. Leyendecker, E.V., "A General Overview of Operation BREAKTHROUGH", Performance Concept in Buildings, National Bureau of Standards Special Publication 361, Vol. 1, 1972, pp. 255-260.

75. Lyons, John, Introduction to Theoretical Linguistics, Cambridge University Press, Cambridge, England, 1968.

76. Lyons, John, Chomsky, 2nd ed., Harvester Press, Sussex, England, 1977.

77. MacAvoy, Paul W. (ed.), OSHA Safety Regulation, American Enterprise Institute for Public Policy Research, Washington, D.C., 1977.

78. Malmberg, Bertil, Structural Linguistics and Human Communication, Academic Press, New York, 1963.

79. Margulis, S.T., "How Environmental Research May Affect the Technical Provisions and Enforcement of Regulations," in Research and Innovation in the Building Regulatory Process, Proceedings of the First NBS/NCSBCS Joint Conference held in Providence, RI, Sept., 1976, P.W. Cooke, ed., National Bureau of Standards Special Publication 473, 1977, pp. 35-54.

80. Masterson, Charles, "The Development of Computer Based Systems for Building Codes," in Research and Innovation in the Building Regulatory Process, Proceedings of the First NBS/NCSBCS Joint Conference held in Providence, RI, Sept., 1976, P.W. Cooke, ed., National Bureau of Standards Special Publication 473, 1977, pp. 369-376.

81. McDaniel, Herman, An Introduction to Decision Tables, John Wiley & Sons, New York, 1968.

82. McDaniel, H., Applications of Decision Tables, A Reader, Brandon/ Systems Press Inc., Princeton, N.J., 1970.

83. McDonagh, N.H., and L.F. Shanley, "SfB Development," in Construction Research International, Proceedings of the Seventh CIB Triennial Congress, Edinburgh, Scotland, Sept, 1977, Vol. 1, Keith Alsop, ed., The Construction Press Ltd., Lancaster, England.

84. Metcalfe, J.W., Subject Classification, Scarecrow Press, New York, 1959.

85. Metzner, John R., and Bruce H. Barnes, Decision Table Language and Systems, Academic Press, New York, 1977.

86. Mill, John Stuart, A System of Logic, 8th Ed., Longmans, Greendand Co. London, 1930.

87. Miller, George A., "The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information," The Psychological Review, Vol. 63, No. 2, pp. 81-97, March 1956.

88. Montalbano, M., "Tables, Flow Charts, and Program Logic," IBM Systems Journal, September, 1962, pp. 51-63.

89. Moriarty, P.J., "Legal Aspects of the Building Regulatory Process," in Research and Innovation in the Building Regulatory Process, Proceedings of the Second NBS/NCSBCS Joint Conference held in Bozeman, MT, Sept., 1977, P.W. Cooke, ed., National Bureau of Standards Special Publication 518, 1978, pp. 381-390.

90. National Building Code (1976 Edition), American Insurance Association, New York, 1976.

91. Newell, A., and H.A. Simon, Human Problem Solving, Prentice Hall, New Jersey, 1972.

92. Noland, J.L., "Formulation of Decision Logic Tables and Their Application to Engineering Specifications," PhD Thesis, Department of Civil Engineering, University of Colorado, 1975.

93. Noland, J.L., J.A. Barker, C.C. Feng, and M.L. Moody, "Constraint Processor (ACICP) for American Concrete Institute Building Code Requirements for Reinforced Concrete (ACI 318-63)," University of Colorado, 1969.

94. Noland, J.L., and C.C. Feng, "Formulation of Decision Logic Tables," Journal of the Structural Division, ASCE, Vol. 97, No. ST1, Proc. Paper 7855, January, 1971, pp. 430-462.

95. Noland, J.L., and C.C. Feng, "The ACI 318 Building Code in Decision Logic Table Format," Journal of the Structural Division, ASCE, Vol. 101, No. ST4, Proc. paper 11220, April 1975, pp. 677-696.

96. Nyman, D.J., S.J. Fenves, and R.N. Wright, "Restructuring Study of the AISC Specification," Civil Engineering Studies, No. SRS 393, University of Illinois, 1973.

97. Nyman, D.J., and S.J. Fenves, "An Organizational Model for Design Specifications," Report R73-4, Department of Civil Engineering, Carnegie-Mellon University, 1973.

98. Nyman, D.J., J.D. Mozer, S.J. Fenves, "Decision Table Formulation of the Load and Resistance Factor Design Criteria," Report R-77-6, Department of Civil Engineering, Carnegie-Mellon University, Pittsburgh, Pennsylvania, 1977.

99. O'Bannon, R.E. Building Department Administration, International Conference of Building Officials, Whittier, CA, 1973.

100. Osgood, C.E., G.J. Suci, P.H. Tannenbaum, The Measurement of Meaning, University of Illinois Press, Urbana, Illinois, 1957.

101. Oster, S., and J.M. Quigley, "Regulatory Barriers to the Diffusion of Innovation" Some Evidence from Building Codes," in Research and Innovation in the Building Regulatory Process, Proceedings of the First NBS/NCSBCS Joint Conference held in Providence, RI, Sept., 1976, P.W. Cooke, ed., National Bureau of Standards Special Publication 473, pp. 113-136.

102. Performance Criteria Resource Document for Innovative Construction, National Bureau of Standards, NBSIR No. 77-1316, 1977.

103. Pfrang, E.O., Manager, The Building Research Division Team, Guide Criteria for the Design and Evaluation of OPERATION BREAKTHROUGH Housing Systems, five volumes, National Bureau of Standards Report 10 200, December 1970.

104. Pollack, S.L., H.T. Hicks, W.J. Harrison, Decision Tables: Theory and Practice, Wiley Interscience, John Wiley and Sons, New York, 1971.

105. Quinne, W.V., Methods of Logic, 3rd ed., Holt, Rinehart, and Winston Inc., New York, 1972.

106. Ranganathan, S.R., Colon Classification, 4th ed., Madras Library Association, Madras, India, 1952.

107. Remmer, N.S. "Regulatory Administration: A Function of Perceived Priorities, Costs and Benefits," in Research and Innovation in the Building Regulatory Process, Proceedings of the Second NBS/NCSBCS Joint Conference held in Bozeman, MT, Sept. 1977, P.W. Cooke, ed., National Bureau of Standards Special Publication 518, 1978, pp. 359-368.

108. Roget, P.M., "Introduction," in Roget's International Thesaurus, Thomas Y. Crowell Company, New York, 1946 (originally published 1852).

109. Sanderson, R.L., Codes and Code Administration, Building Officials Conference of America, Inc., Chicago, 1969.

110. Schoen, R., R. Hirshberg, and J. Weingart, New Energy Technologies for Buildings: Institutional Problems and Solutions, Ballinger Publishing Company, Cambridge, Mass, 1975, pp. 171-200.

111. Schumacher, Helmut, and Kenneth C. Sevcik, "The Synthetic Approach to Decision Table Conversion," Communications of the ACM, Vol. 19, No. 6, June, 1976, pp. 343-351.

112. Schwayder, K., "Extending the Information Theory Approach to Converting Limited Entry Decision Tables to Computer Programs," Communications of the ACM, Vol. 17, No. 9, September, 1974, pp. 532-537.

113. Seaberg, P., "Decision Table Formulation of the Specification for the Design for Cold-Formed Steel Structural Members," Department of Civil Engineering, University of Wisconsin, Milwaukee, Wisconsin, July, 1971.

114. Seiss, C.P., "Research, Building Codes, and Engineering Practice," ACI Journal, Vol. 31, No. 10, May 1960, Proceedings Vol. 56, pp. 1105-1122.

115. Simpson, G.G., C.S. Pittendrigh, and L.H. Tiffany, Life: An Introduction to Biology, Harcourt, Brace, and Co., New York, 1957.

116. Simpson, John A., "The Dialectic of Measurement," unpublished manuscript, National Bureau of Standards, November, 1975.

117. Sneath, P.H.A., and R.R. Sokal, Numerical Taxonomy, W.H. Freeman and Company, San Francisco, Ca., 1973.

118. Sokal, Robert R., "Classification: Purposes, Principles, Progress, Prospects," Science, Volume 185, Number 4157, September 27, 1974.

119. Specification for the Design, Fabrication, and Erection of Structural Steel for Buildings, American Institute of Steel Construction, New York, 1969.

120. Sprunt, R., "Building Knowledge and Building Law," Journal of Architectural Research 4/3, December 1975.

121. Standard Building Code (1976 Edition), Southern Building Code Congress Internationl, Birmingham, Alabama, 1976.

122. Stevens, R.F., and L. Monument, "A Study of Coding and Data Coordination for the Construction Industry: Structural Engineering Design," Current Paper 5/69, Building Research Station, Garston, Watford, WD2 7JR, England.

123. Sussman, J.M., R.D. Logcher, and H.C. Stotland, "SPECS - A Specification Production Computer System," Journal of the Structural Division, ASCE, Vol. 97, No. ST1, Proc. Papaer 7844, January 1971, pp. 425-438.

124. Tavis, R.L., and J.W. Melin, "Use of Technical Analysis in Editing," Civil Engineering Studies No. SRS 473, University of Illinois, Urbana, Illinois, 1980.

125. Tichy, M., "Codes for Structural Analysis and Design," BUILD International, May/June 1971.

126. Uniform Building Code (1976 Edition), International Conference of Building Officials, Whittier, California, 1976.

127. Uniform Plumbing Code (1973 Edition), International Association of Plumbing and Mechanical Officials, Los Angeles, California, 1973.

128. Ventre, F.T., "SOCIAL CONTROL OF TECHNOLOGICAL INNOVATION: The Regulation of Building construction", unpublished doctoral dissertation, Massachusetts Institute of Technology, 1973.

129. Ventre, F.T., "Decision Aiding Communications in the Regulatory Process: The Partisan Uses of Technical Information," in Research and Innovation in the Building Regulatory Process, Proceedings of the First NBS/NCSBCS Joint Conference held in Providence, RI, Sept., 1976, P.W. Cooke, ed., National Bureau of Standards Special Publication 473, 1977, pp. 203-223.

130. Vickery, B.C., Classification and Indexing in Science, Butterworth & Co., London, 1975.

131. Websters New International Dictionary, Second edition, G. & C. Merriam Company, Springfield, Mass., 1946.

132. Wright, J.R., "Performance Criteria in Building," Scientific American, Vol. 224, No. 3., March 1971, pp. 17-25.

133. Wright, R.N., and A.H.-S. Ang, "A Consistent Basis for Functional and Ultimate Criteria," Performance Concept in Buildings, Proceedings of the Joint RILEM-ASTM-CIB Symposium, National Bureau of Standards Special Publication 361, Volume 1, 1972, pp. 181-190.

134. Wright, R.N., L.T. Boyer, and J.W. Melin, "Constraint Processing in Design," Journal of the Structural Division, ASCE Vol. 97, No. ST1. Proc. Paper 7856, January, 1971, pp. 481-494.

135. Wright, R.N., J.R. Harris, J.W. Melin, C. Albarran, "Technology for the Formulation and Expression of Specifications, Volume III: Technical Reference Manual," Civil Engineering Studies No. SRS 425, University of Illinois, Urbana, Illinois, 1975.

136. Wright, R.N., D.J. Nyman, and S.J. Fenves, "Progress Report on Restructuring Study of Steel Specifications," (report to the American Institute of Steel Construction), Department of Civil Engineering, University of Illinois, 1971.

137. Workman, J.R., "Aristotle," Encyclopedia Americana, Vol. 2, 1973.

138. Yngve, V., "Computer Programs for Translation," Scientific American, Vol. 206, No. 6, June, 1962, page 68.

139. Yntema, Douwe B, and Gayle E. Mueser, "Remembering the Present States of a Number of Variables," Journal of Experimental Psychology, Vol. 60, No. 1, pp. 18-22, 1960.

140. Yokel, F.Y., and N.F. Somes, "Philosophy and Scope of Structural Performance Criteria," Performance Concept in Buildings, National Bureau of Standards Special Publication 361, Vol. 1, 1972, pp. 261-266.

141. Zarnikow, W.E., "Keeping up with Codes and Standards Information: A Specification Writers Dilemma," Engineering Extension, Iowa State University, November, 1976.

GLOSSARY

Action - the lower portion of a decision table; it contains the statements used to assign a value to a datum.

Application requirement - a requirement that depends conditionally on other requirements.

Argument - a classifier of a provision.

Attribute - a quality required of a THING; usually used in the form "performance attribute."

Basic categories - types of facets and fields that are commonly useful for classifying provisions in standards; developed in this report for building standards only.

Basic requirement - a requirement that assigns one REQUIRED QUALITY for one THING and does not directly depend, conditionally or unconditionally, upon any other requirements.

Branch - for an information network, a link connecting two nodes directed from the ingredient node to the dependent node; for a classifier tree, one continuous path, including nodes, from the root to a terminal node.

Brother - see Sibling

Category - see basic category.

Characteristic - a quality used in classing.

Class, noun - a group of things possessing common characteristics

Class, verb - to assign things to a class

Classification - a system of classes grouped and ordered according to some principles; also, the process of establishing such a system.

Classifier - the name of a class.

Classify - to make or conceive a class or classes and to arrange the classes in a classification.

Cluster - a group of provision references associated with a heading in an index.

Collective exhaustion - a principle that all sibling classes together contain all things in the parent class; one of the logical principles of classification.

Condition - the upper portion of a decision table used to express logical statements.

Conditional Order - a linear order for the nodes in an ingredience network that begins with the final dependent and then proceeds to define each ingredient after its own dependent.

Cousin - for a classifier, any classifier from the same facet that is not either a predecessor or a descendent of the classifier in question.

Criterion - a requirement that uses an explicit comparison, or test, in order to determine its value.

Cumulative requirement - a datum that depends unconditionally on one or more other requirements and that does not address a THING or a REQUIRED QUALITY except through its ingredient requirements.

Datum - (plural datums, data items) a unit of information used to evaluate compliance with a standard: each requirement corresponds to one datum and such a datum's value is either "satisfied" or "violated;" each variable established by one or more determinations is a datum, and each input variable is a datum.

Decision table - a set of rules specifying certain actions to be executed based on a specific set of conditions.

Dependence (global) - for a datum, that portion of the information network that is located on paths composed of branches pointing away from the node representing that datum.

Dependent - a datum whose value may depend directly on the value of the datum in question.

Derived datum - any datum with ingredients; a derived datum represents the information in some part of one or more provisions.

Descriptive quality - a characteristic used to classify THINGS (the subjects of requirements).

Descendent - for a classifier tree, any classifier located on a branch in common with the classifier in question, but further removed from the root.

Determination - a provision that is not a requirement; more than one determination may be represented by the same datum.

Direct order - a linear order for the nodes in an ingredience network that begins with an input node and then proceeds to define each dependent only after each of its ingredients are defined.

Entry - the right-hand portion of a condition or action in a decision table; the matrix of entries corresponds to the rules.

Explicit entry - a condition entry that is Y (true, or T) or N (false, or F); contrast with implicit and immaterial.

Facet - in library science, facets are "various hierarchies that can occur in the classification of a subject field" [130]; in this study each grouping in a facet must satisfy the logical principles of classification, thus a facet is a true tree, not simply hierarchy; in this study, the largest portion of a classification that maintains the logical principles of classification.

Faceted classification - a classification in which each field of the classification is divided into one or more facets.

Field - the largest unit in a classification; each field is considered as an independent subject area.

Global - see dependence and ingredience.

Heading - a title used to indicate the scope of the provision(s) that it denotes; composed of classifiers.

Hierarchy - a division of classes at several levels in which each class in a level has a smaller scope than the classes in the next higher level.

Immaterial entry - a condition entry used to indicate that the condition value may be true or false, i.e., that the condition has no bearing on the rule; it is symbolized by ".".

Implicit entry - a condition entry used to indicate that the condition value is predetermined, for the rule in question, by the explicit entry for some related condition for that rule; implicit true is symbolized by "+" and implicit false is symbolized by "-".

Information network - the assembly of nodes and branches in the form of a linear graph that is used to represent the precedence of evaluation for all the datums in a standard.

Ingredience (global) - for a datum, that portion of the information network that is located on paths composed of branches pointing toward the node representing that datum.

Ingredient - a datum whose value may be required to establish the value of the datum in question.

Input datum or node - a datum with no ingredients.

Kernel - a simple declarative sentence that forms all or a part of the underlying structure of a sentence.

Limit state - an event that may cause the loss of a performance attribute, either by its occurrance or its amplitude.

Logical principles of classification - see mutual exclusion and collective exhaustion.

Mixed requirement - a requirement that assigns one or more REQUIRED QUALITIES to one or more THINGS and also depends directly, conditionally or unconditionally, on other requirements.

Multiple requirement - a requirement that is equivalent to more than one basic requirement.

Mutual exclusion - a principle that no one thing will be in more than one class from a set of siblings; one of the logical principles of classification.

Node - a point in an information network that represents one datum.

Nuclear tree - one set of sibling classes connected to their parent class; the smallest element of a classification that maintains the logical principles.

Objectives (of organization) - the qualities relating headings and their subordinate provisions established as goals for a good organization and used to establish various criteria; specified and defined in section 1.2 (Relevant, Meaningful, Unique, Complete, Graded, Progressive, Intelligible, Minimal, and Even).

Organization - the scope and the arrangement of provisions within a standard.

Organizational tree - a tree of classifiers formed by appending nuclear trees from various facets and fields together with provisions entered at the appropriate points; may be converted to an outline by converting the classifers to headings.

Outline - the headings that constitute an arrangement (i.e. a grouping and ordering) of the provisions in a standard.

Output datum or node - a datum with no dependents.

Performance attribute - a REQUIRED QUALITY that satisfies a basic need of the user without prescribing or presuming a solution scheme.

Phenomenon - an event that is significant for the objective measurement of a performance attribute.

Physical entity - an abstract term that includes objects, materials, and systems of objects.

Predecessor - for a classifier tree, any classifier located on a branch in common with the classifier in question, but closer to the root.

Property - a descriptive quality.

Provision - a rule for judging compliance with a standard; all provisions are divided into two classes: requirements and determinations; may be expressed as all or part of one or more sentences, but commonly is a single sentence.

RCEC - "Requirement-Criteria-Evaluation-Commentary" - a common format for performance standards, to wit: each requirement has one or more criteria, and each criterion has an evaluation and a commentary.

REQUIRED QUALITY - the predicate in the model of a requirement idealized for purposes of relevant classification.

Requirement - a provision that may directly indicate compliance with some portion of a standard; its value may be characterized as "satisfied" or "violated."

Root - the classifier at the head of a facet (a classifier tree); a class with no parent class.

Rule - one column of a decision table that specifies which action is to be executed when the conditions have a prescribed set of values.

Sibling - one of a group of classes that have the same parent; normally a set of siblings satisfies the logical principles of classification.

Spanning tree - a tree that includes all the nodes in a network and just enough branches to connect the nodes without any closed meshes.

Standard - a set of provisions; includes code and specification, for this study.

Step siblings - classes that become siblings by virtue of their placement in an organizational tree or an outline, but which were not originally siblings.

Stub - the left-hand portion of the conditions or actions in a decision table; contains the logical statement for a condition or the executable statement for the action.

Synthetic Requirement - a requirement that depends conditionally on other requirements and is functionally equivalent to a new basic requirement.

Terminal - see output.

THING - the subject in the model of a requirement idealized for purposes of relevant classification.

Tree - a network of nodes and branches with no closed meshes and with a single root node; for this study, each node is a class.

Value - the quantity or quality that represents a datum.

APPENDIX

COMPUTER PROGRAM "OUTDEX"

OUTDEX is a FORTRAN program to read and store data concerning the classification of provisions, display that information in many forms, produce an alphabetical index, enter provisions in any given outline of headings, and interactively create outlines. The program is a prototype, established primarily to develop and test algorithms for indexing and outlining. Thus, it is not truly suitable for general use, and this documentation is brief and incomplete.[1/]

The program consists of an MAIN routine and 22 subroutines along with a BLOCK DATA subprogram and a procedure element, SPECS, that contains several specification statements used in many of the subroutines. The 25 elements include approximately 2000 source language statements and are in one program file on the UNIVAC 1108 at the National Bureau of Standards. The program also makes use of a package of subroutines known collectively as PARSE. [2/] PARSE is used to examine input records and makes possible the use of somewhat format-free and problem-oriented input language.

The subroutine can be grouped into three functional areas, as shown in figure A1. Subroutine linkages for each of the three areas are shown in figures A2, A3, and A4. Figures A5 and A6 show the overall flow of control in the program. Figure A6 is essentially a "grammar" for the principal command language. The conventions used in figure A6, and some subsequent figures are as follows:

- <PRINT> implies a test of the input record to match the word "PRINT". All other tests for words in upper case letters are similar. The test is passed if the first three letters of any word are matched.

- <fixed> is a test for any integer number.

- <end> is a test for the end of an input record.

- the right hand path is taken if the test is passed while the downward path is taken if it is failed.

- the diagram branches are always oriented downward or to the right unless otherwise indicated by an arrowhead.

[1/] OUTDEX unintentionally uses terms that imply male characteristics for the classifier data. To avoid confusing potential users of the software, the text and figures in this appendix use the same terminology as OUTDEX. These terms, with their equivalents from the main body of this report in parentheses, are: son (child), brother (sibling), and extended brother (cousin). Future versions of the software will use the genderless terms.

[2/] PARSE was developed at the Civil Engineering Systems Laboratory of the University of Illinois, Urbana, Illinois.

Figure A.7 shows the overall flow of control for the INPUT subroutine, and figure A.8 through A.11 give more detail for various portions. The most import restrictions on the input are the following:

- The classifiers must be entered before any other data.

- The classifiers should not be ordered other than in ascending numerical (absolute value) order.

- The provision numbers must be entered in ascending numerical (absolute value) order.

- The provision number must be identical to the datum number for the same provision in the information network file, and

- The provision numbers must be entered before the provision titles.

These restrictions allow the program to read the input file for the information network program [52], which is a timesaver. Provision and classifier numbers may be signed for several purposes. Negative provision numbers indicate Determinations (non-Requirements). Negative argument numbers indicate that the classifier is not to be considered an argument in the outlining mode. Negative classifier numbers may be used to indicate the root of a facet that is not the root of a field; such classifiers are referred to as "transparent" in the program.

Figure A.12 gives the sequence of calculation in subroutine TOPANL, which includes all the preliminary analysis and data restructuring that is done prior to execution. Figures A.13 through A.17 present some detail for these calculations. Most of these algorithms are common routines for traversing networks and are not unique to this program.

Subroutine PRINT is used to display the input information in several forms and the data calculated in subroutine TOPANL. There are nine possible options for this display, as listed below. No flow charts are provided for any of these printing options. The options are selected in MAIN, and the numbers in parentheses shown on various branches of figures A.6 correspond to the numbers in the list below:

1) List the data number for each provision entered and the corresponding data description. The data number is positive for Requirements and negative for Determinations.

2) List the number and title for each classifier entered. The number is negative if the classifier is not used in an outlining mode for Requirements. The title is preceded by an asterisk for all transparent classifiers. Blank lines are printed between classifiers that are not consecutive.

3) List the number and description for each provision, as in 1), but also list the arguments, by number and title, immediately beneath each provision.

4) List the number and title for each classifier and the number and description of each provision in the scope list immediately beneath it.

5) List the basic data calculated in subroutine TOPANL: the number of provisions and classifiers indexed and outlined, the number of roots, foster parents, facets (transparent classifiers), context classifiers, and the number of indexing and outlining associations between provisions and classifiers.

6) List each provision and the fields with which it is associated.

7) List the number and title of each classifier, indented according to its level, and the Requirements in its scope list.

8) List the number and title of each classifier, indented according to its level, and the Determinations in its scope list.

9) List the number and title of each classifier, indented according to its level, and all provisions in its scope list.

Option 2) is a printing of the classification system without considering the secondary hierarchy of foster parents. Option 4) is an index ordered according to the input order of the classifiers. Options 7), 8), and 9) are simple outlines printed one facet at a time; these outlines satisfy only the "local relevance" criterion discussed in section 6.2.2.

Subroutine INDEX produces an index ordered by the alphabetical sequence of the classifiers. Only classifiers with a non-empty scope list are used as headings in the index. Figure A.18 shows the overall logic of the subroutine. The alphabetical sorting routine is very simple-minded and requires a relatively large amount of processing time. Note that the algorithm used for subdividing large clusters of provisions at a single heading is not shown; see the discussion in section 6.1.3.

Subroutine SORT is used for two operations: determining all provisions that have a given set of classifiers among their arguments or determining all provisons that have no arguments among a given set of classifiers. Figure A.19 shows the logic for the subroutine: "match" is the former operation and "exclude" is the latter.

OUTLIN is the principal subroutine for the generation of an outline. Figure A.20 shows the overall flow of control for this process. The subroutine is designed to build an outline interactively, store the organizational tree thus created to allow direct editing offline and then reprocess the edited organizational trees in a batch type of operation. OUTDEX does not generate an organizational tree, as some earlier computer programs [96, 97, 135] have done. No algorithm has been developed that can successfully generate an organizational tree from the complex classification systems developed in this work.

The objective of the interactive outline building routine is to harness the power of computers to: 1) sort through all the linkages of provisions and classifiers and 2) make the checks for logic and relevance. In this way the computer can give direction to the process. The user must decide how to append classifier trees and where to break the logical rules implicit in the classification. This creative power of the user is the essential element in creating outlines. The computer can make that creative power more effective by relieving the user of countless mundane checks that must be made. Figure A.21 describes the interactive process, using the same notation for the command language grammar as was described for figure A.6.

The interactive routine is worthy of a more detailed discussion than much of the program, because the philosphy is new. The basic concept is that the user guide the organizational tree one branch at a time, adding one level to the branch in each step. The user does this by specifying the parent of those classifiers that make up the new level. The current branch is extended by adding the first son of that parent. At the same time, OUTDEX stores the next brother of the first son as the first node on a new branch at that level.

OUTDEX then searches for provisions to enter on the current branch of the organizational tree, printing them as they are found. The program also searches for provisions that potentially could be entered on the current branch if more classifiers were added to it. Although there are several more complicated options, as shown in figure A.21, the basic logic for the next move of the program is illustrated best by the following decision table.

Potential future match exists for this branch	*	Y	N	N
Provision entered at this heading	*	.	Y	N
**				
Await command to append more classifiers	*	X		
Create new branch by getting next brother	*		X	
Replace current heading with next brother	*			X

The user may request the program to display each of the potentially matching provisions, along with the arguments necessary to allow it to be entered in the outline. This provides guidance as to which classifiers should be appended next.

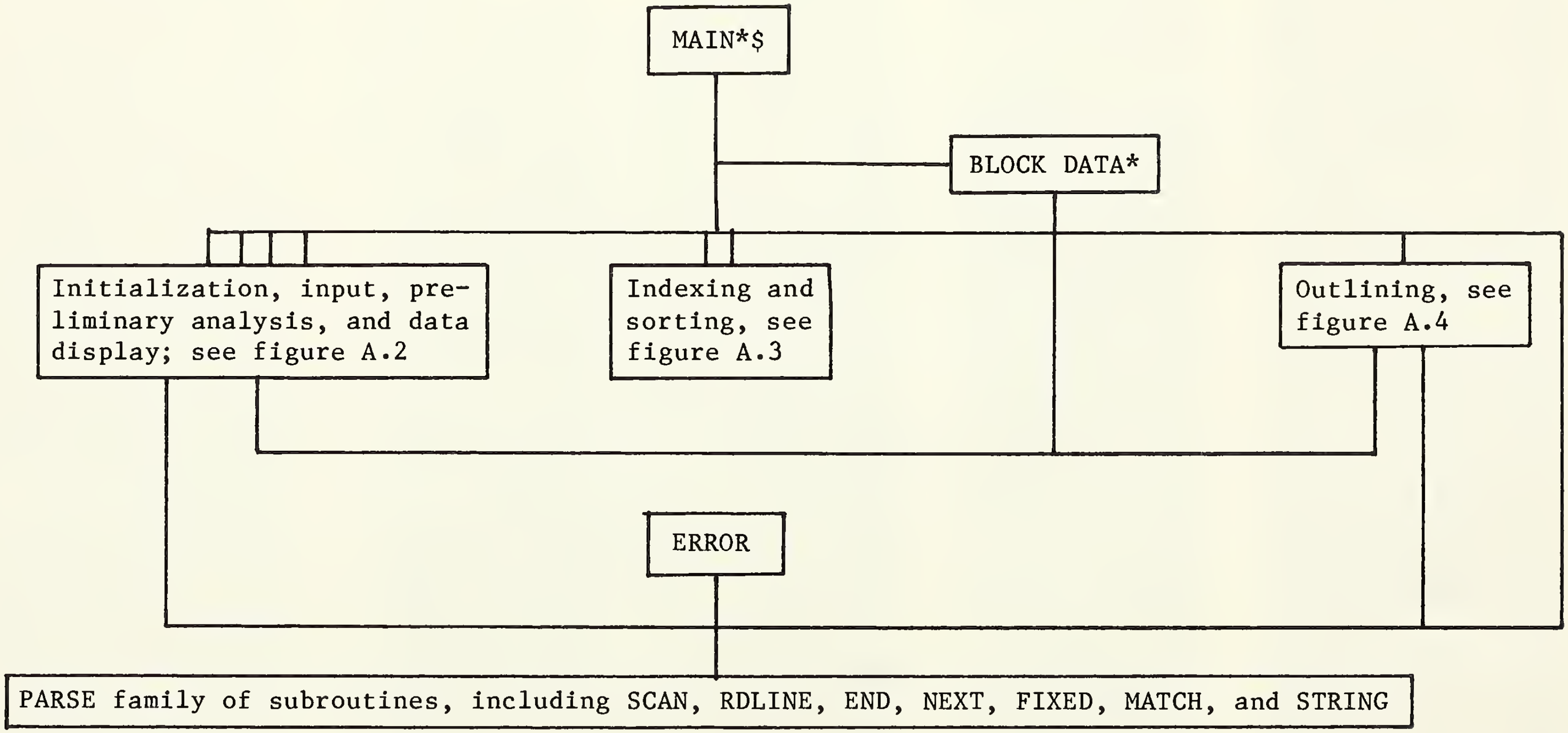
An option allows the program to skip the search for potentially matching provisions at high levels of the organizational tree, that is, those levels with very few classifiers on the branch. This is useful because it is likely that such provisions would always be found at high levels. The "DEPTH" parameter referred to on figure A.21 is the level beyond which the search is made. A related option directs the program not to delete empty headings (those headings with no provision matching and no provisions potentially matching). The "SAVE" parameter automatically saves headings for those levels at which no search for potential provisions is made. It may be selectively turned on and off by the user at the lower levels.

When the program attempts to use the next brother for a heading and finds none, the normal situation is to return to a higher level and use the next brother of the current heading at the higher level. However, the program allows the user to "hold" the current level for entry of more classifiers. Holding the lower level and adding more classifiers allows the user to make "brothers" of classifiers that are not brothers in any facet of the overall classification, thus breaking the logic inherent in the classification. It is also possible to break the logic of the classification by entering a single classifier rather than all of the sons of a classifier. This is done by using the keyword "ROOT" in the command, as shown in figure A.21.

Figures A.22 and A.23 show, respectively, the subroutines that search for provisions to enter on a specific branch and for provisons with potential for entry at a lower level on the branch. Both subroutines use the same algorithms for testing the provisions for entry into the outline. The logical criteria are contained in subroutine LOGIC, shown in figure A.24 and the full relevance criterion is contained in subroutines FULL, DIRECT, and INDRCT, shown in figures A.25 through A.31. The criteria for testing provisions are discussed in some detail in section 6.2.2, so less detail is offered here.

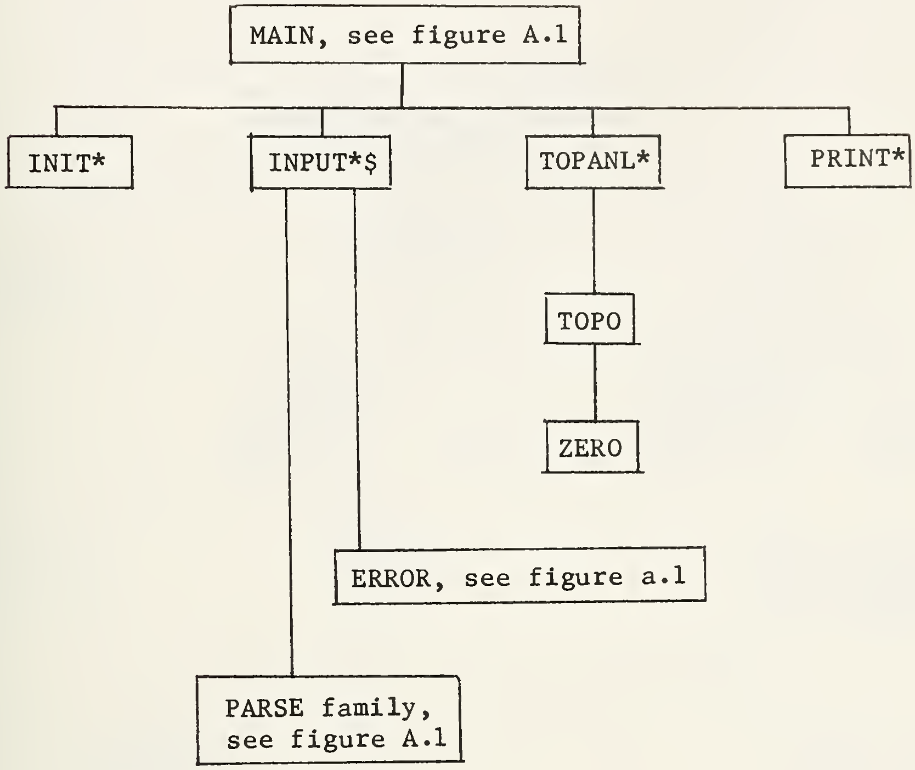
Subroutine TEST (figure A.22) is used in both the interactive mode of outline creation and the "batch" mode of processing a given organizational tree, whereas subroutine POTEN (figure A-23) is in the interactive mode only. The criteria UNIQUE, GRADED, and FULL for testing provisions are sequentially applicable, that is, if GRADED is applicable then UNIQUE must be, and if FULL is applicable then GRADED must be. Testing the FULL criterion involves significantly more execution than the simpler criteria, and its use would serve no purpose for classification systems that have no more than one facet in a field.

The user is also able to selectively remove and replace classifiers from active consideration in the provision searching routine by using the DROP and RETRIEVE commands. This allows considerable flexibility in meeting the logical criteria, albeit in a rather covert fashion. As shown in figure A.32, single classifiers, sets of brothers, or entire facets may be addressed with a single command.


MAIN*$
BLOCK DATA*
Initialization, input, preliminary analysis, and data display; see figure A.2
Indexing and sorting, see figure A.3
Outlining, see figure A.4
ERROR
PARSE family of subroutines, including SCAN, RDLINE, END, NEXT, FIXED, MATCH, and STRING

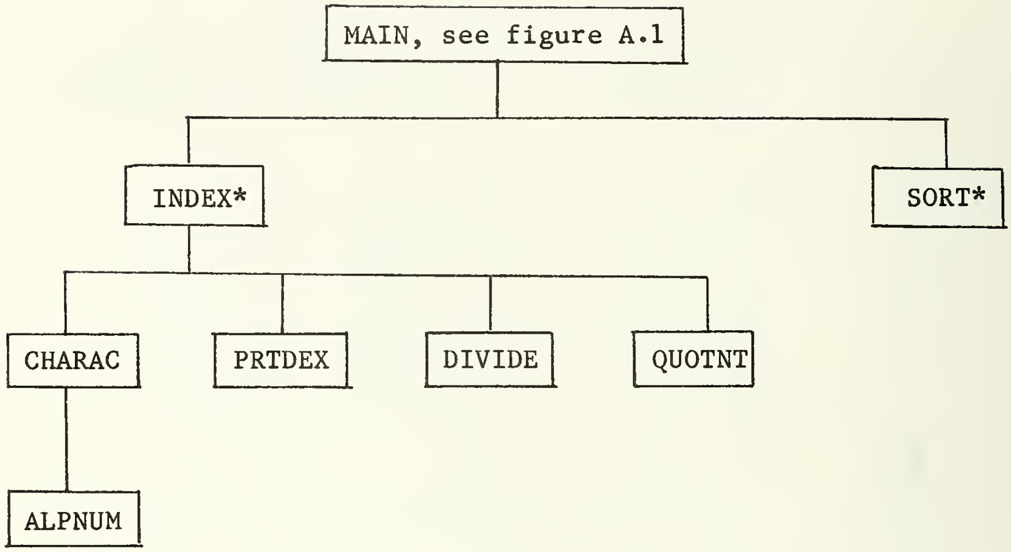
* Routine includes OUTCOM, a set of specification statements in the element SPECS
$ Routine includes PARCOM, a set of specification statements in the element SPECS

Figure A.1 Overall Program Structure



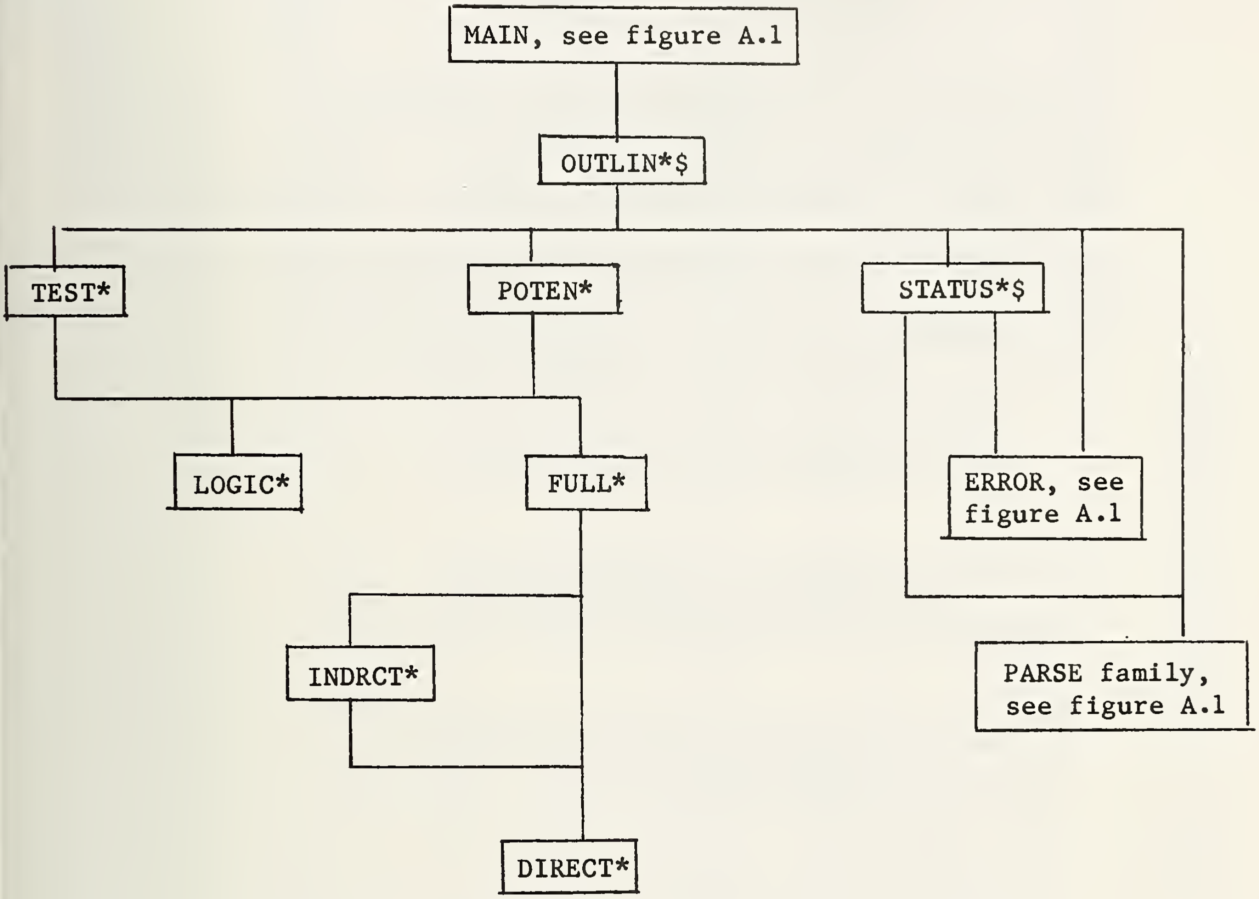
* Routine includes OUTCOM
$ Routine includes PARCOM

Figure A.2 Subroutine Linkage for Preliminary Portion



* Routine includes OUTCOM

Figure A.3 Subroutine Linkage for Indexing and Sorting Portion



* Routine includes OUTCOM
$ Routine includes PARCOM

Figure A.4 Subroutine Linkage for Outlining Portion

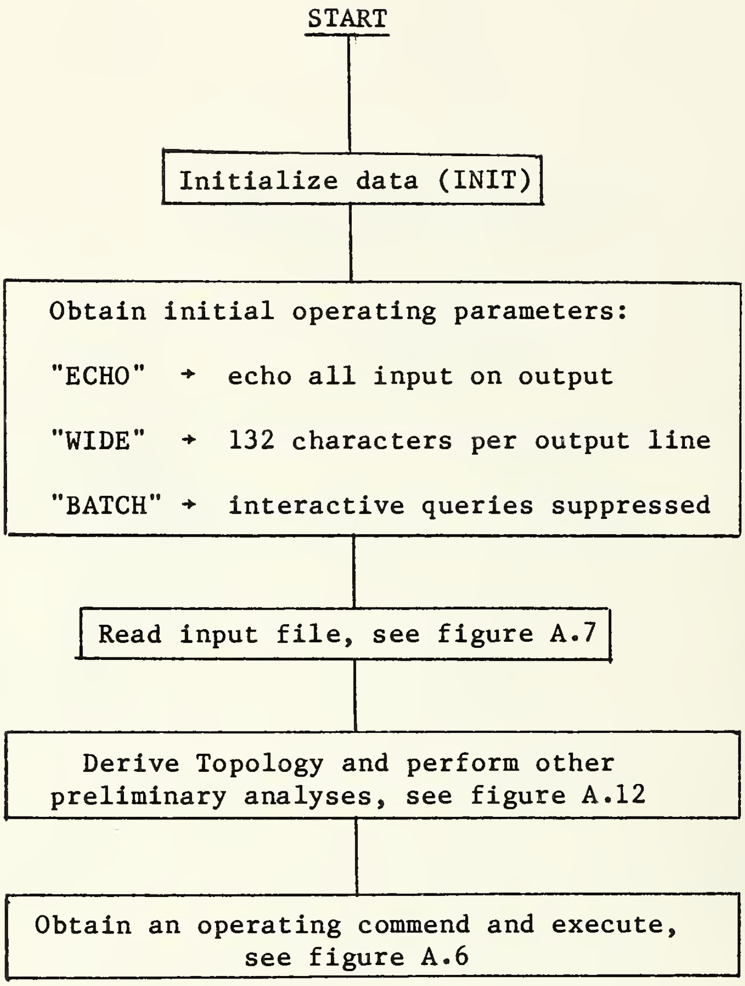

START
Initialize data (INIT)
Obtain initial operating parameters:
"ECHO" → echo all input on output
"WIDE" → 132 characters per output line
"BATCH" → interactive queries suppressed
Read input file, see figure A.7
Derive Topology and perform other preliminary analyses, see figure A.12
Obtain an operating commend and execute, see figure A.6

Figure A.5 MAIN Program Flow of Control

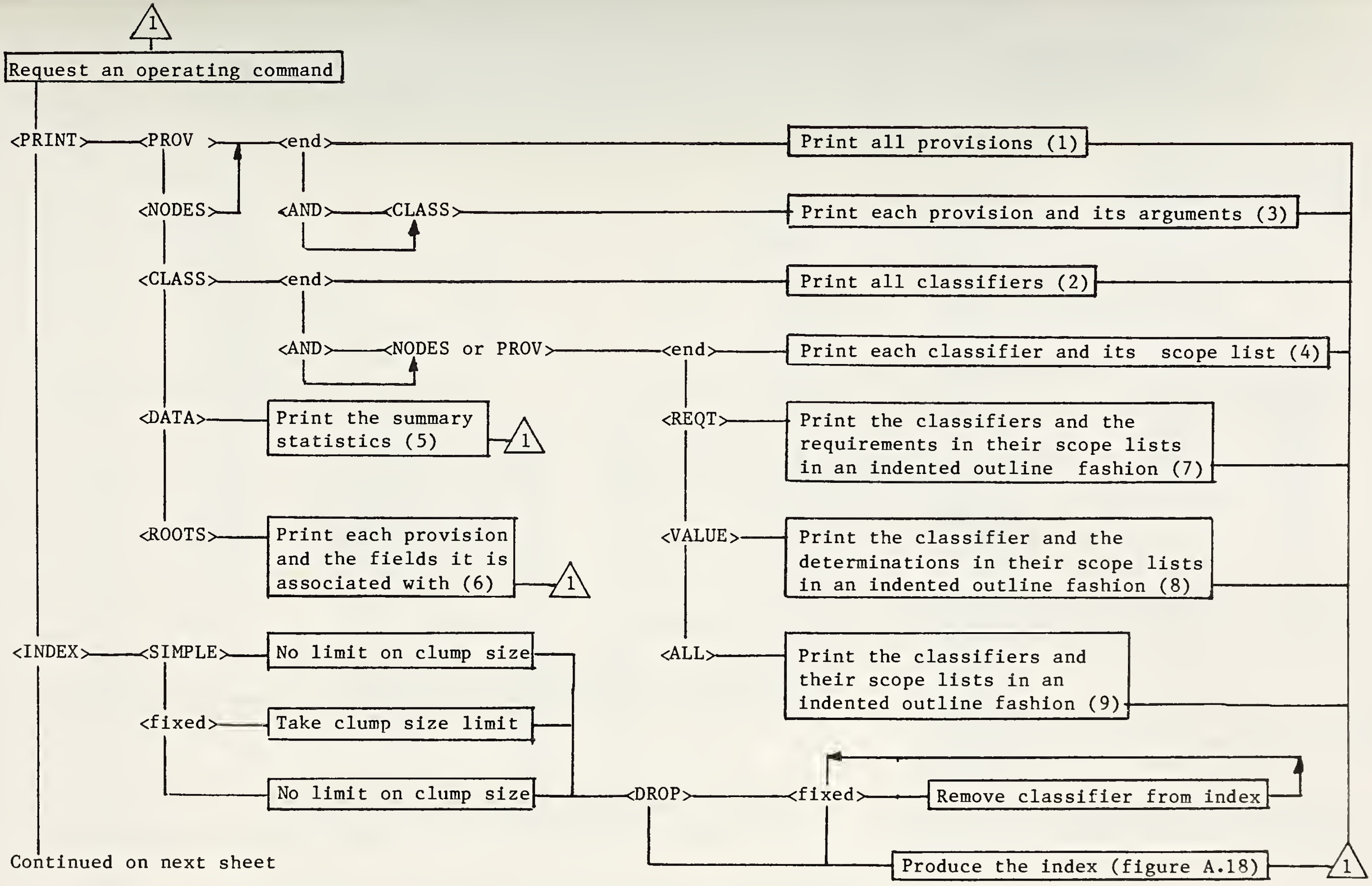

1
Request an operating command
<PRINT>
<PROV >
<end>
Print all provisions (1)
<NODES>
<AND>
<CLASS>
Print each provision and its arguments (3)
<CLASS>
<end>
Print all classifiers (2)
<AND>
<NODES or PROV>
<end>
Print each classifier and its scope list (4)
<DATA>
Print the summary statistics (5)
1
<REQT>
Print the classifiers and the requirements in their scope lists in an indented outline fashion (7)
<ROOTS>
Print each provision and the fields it is associated with (6)
1
<VALUE>
Print the classifier and the determinations in their scope lists in an indented outline fashion (8)
<INDEX>
<SIMPLE>
No limit on clump size
<ALL>
Print the classifiers and their scope lists in an indented outline fashion (9)
<fixed>
Take clump size limit
No limit on clump size
<DROP>
<fixed>
Remove classifier from index
Produce the index (figure A.18)
1
Continued on next sheet

Figure A.6 Operating Command Language (MAIN)
---concluded on next page---

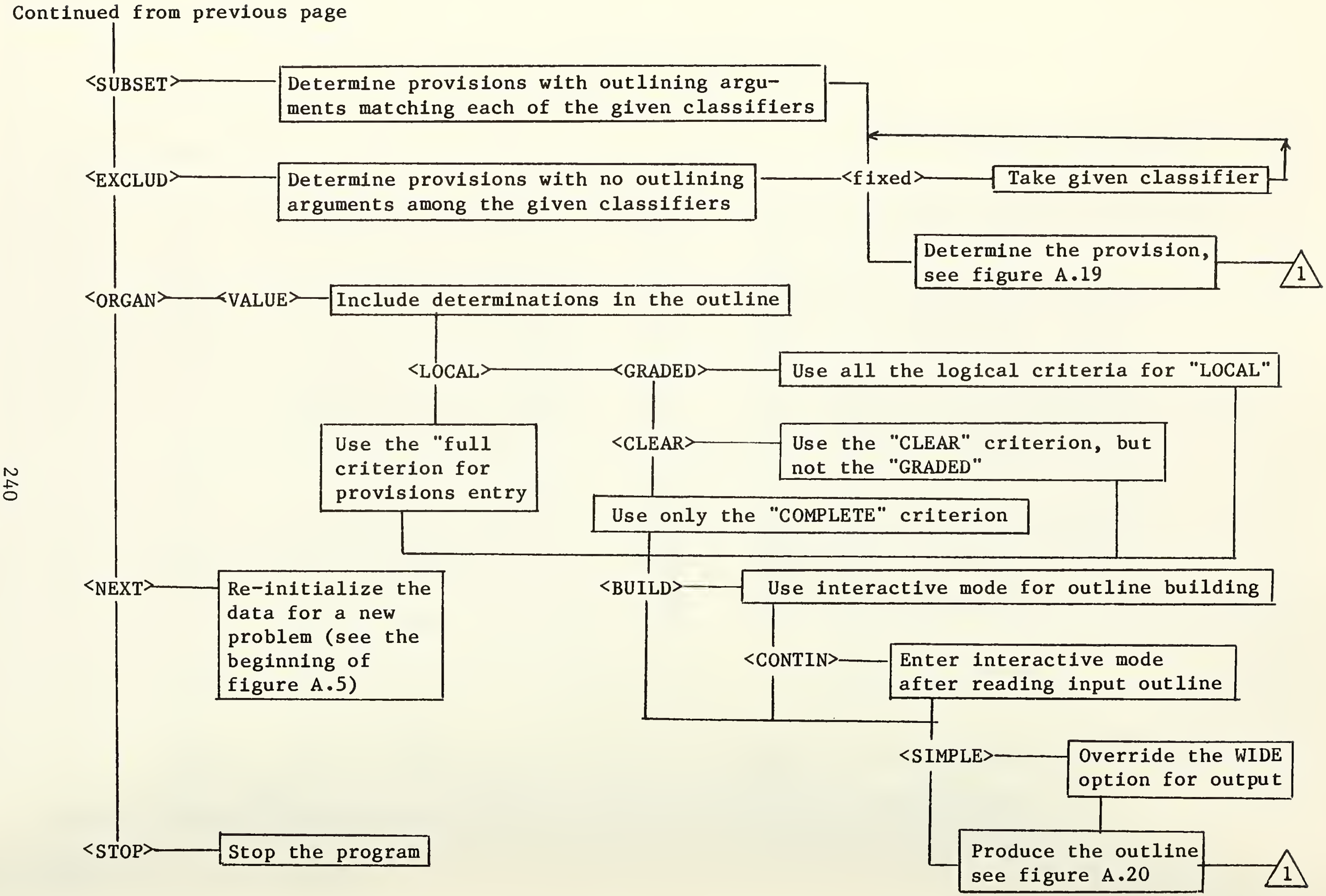

Continued from previous page
<SUBSET>
Determine provisions with outlining arguments matching each of the given classifiers
<EXCLUD>
Determine provisions with no outlining arguments among the given classifiers
<fixed>
Take given classifier
Determine the provision, see figure A.19
1
<ORGAN>
<VALUE>
Include determinations in the outline
<LOCAL>
<GRADED>
Use all the logical criteria for "LOCAL"
Use the "full criterion for provisions entry
<CLEAR>
Use the "CLEAR" criterion, but not the "GRADED"
Use only the "COMPLETE" criterion
<NEXT>
Re-initialize the data for a new problem (see the beginning of figure A.5)
<BUILD>
Use interactive mode for outline building
<CONTIN>
Enter interactive mode after reading input outline
<SIMPLE>
Override the WIDE option for output
Produce the outline see figure A.20
1
<STOP>
Stop the program

Figure A.6 Operating Command Language (MAIN)
--continued from previous page--

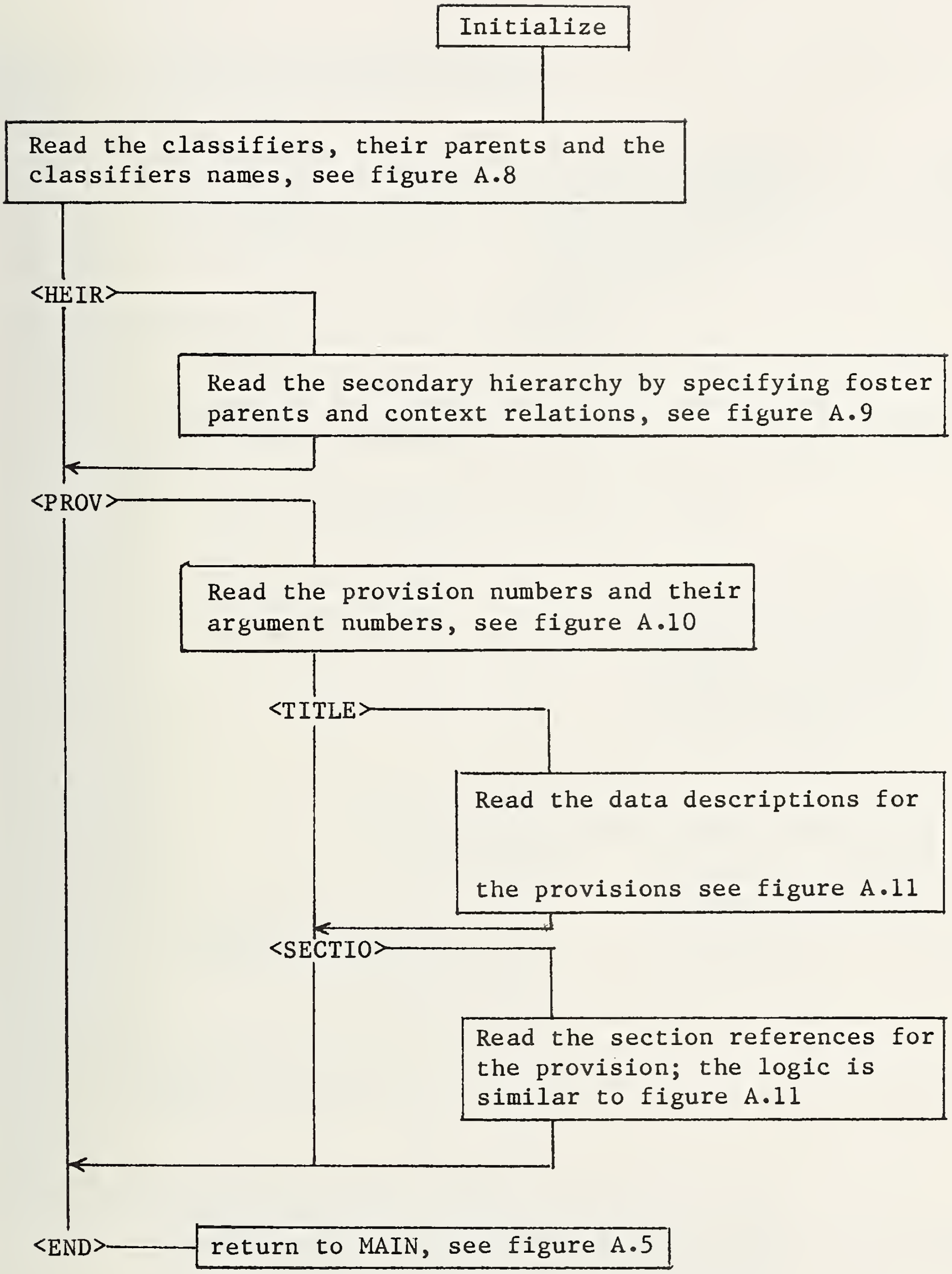

Initialize
Read the classifiers, their parents and the classifiers names, see figure A.8
<HEIR>
Read the secondary hierarchy by specifying foster parents and context relations, see figure A.9
<PROV>
Read the provision numbers and their argument numbers, see figure A.10
<TITLE>
Read the data descriptions for the provisions see figure A.11
<SECTIO>
Read the section references for the provision; the logic is similar to figure A.11
<END>
return to MAIN, see figure A.5

Figure A.7 INPUT Flow of Control

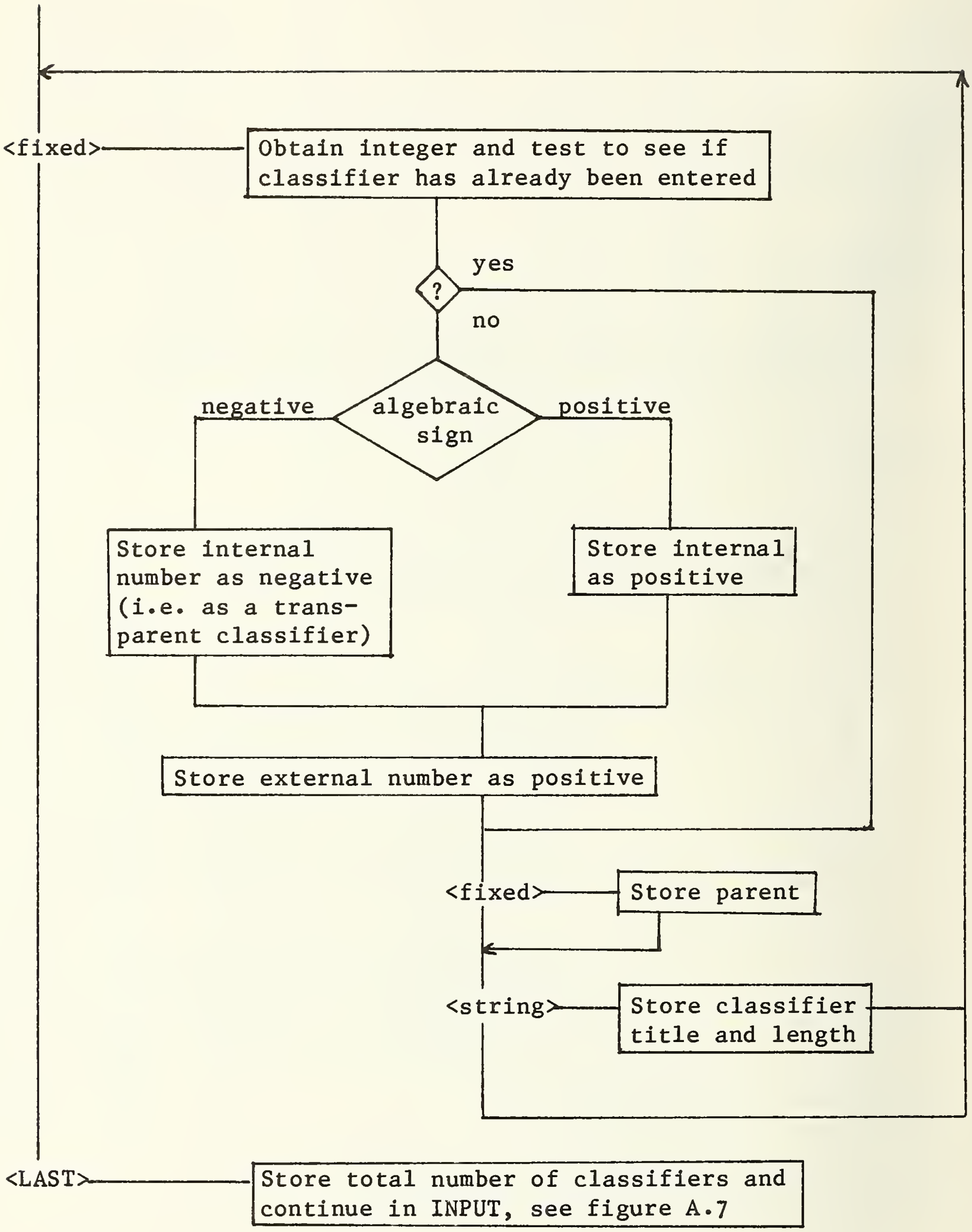

<fixed>
Obtain integer and test to see if classifier has already been entered
?
yes
no
negative
algebraic sign
positive
Store internal number as negative (i.e. as a transparent classifier)
Store internal as positive
Store external number as positive
<fixed>
Store parent
<string>
Store classifier title and length
<LAST>
Store total number of classifiers and continue in INPUT, see figure A.7

Figure A.8 Entry of Classifiers and Parents (INPUT)

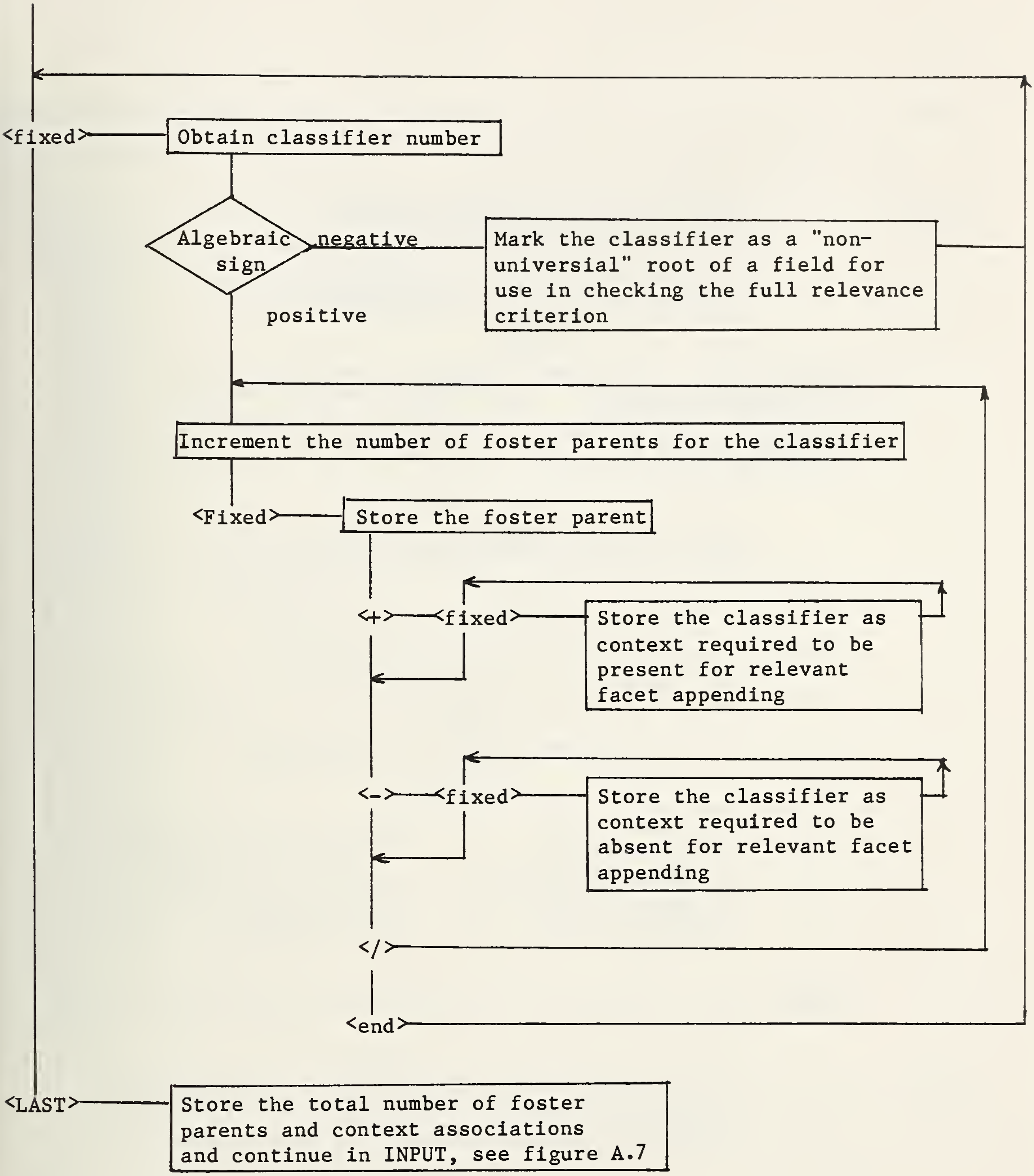
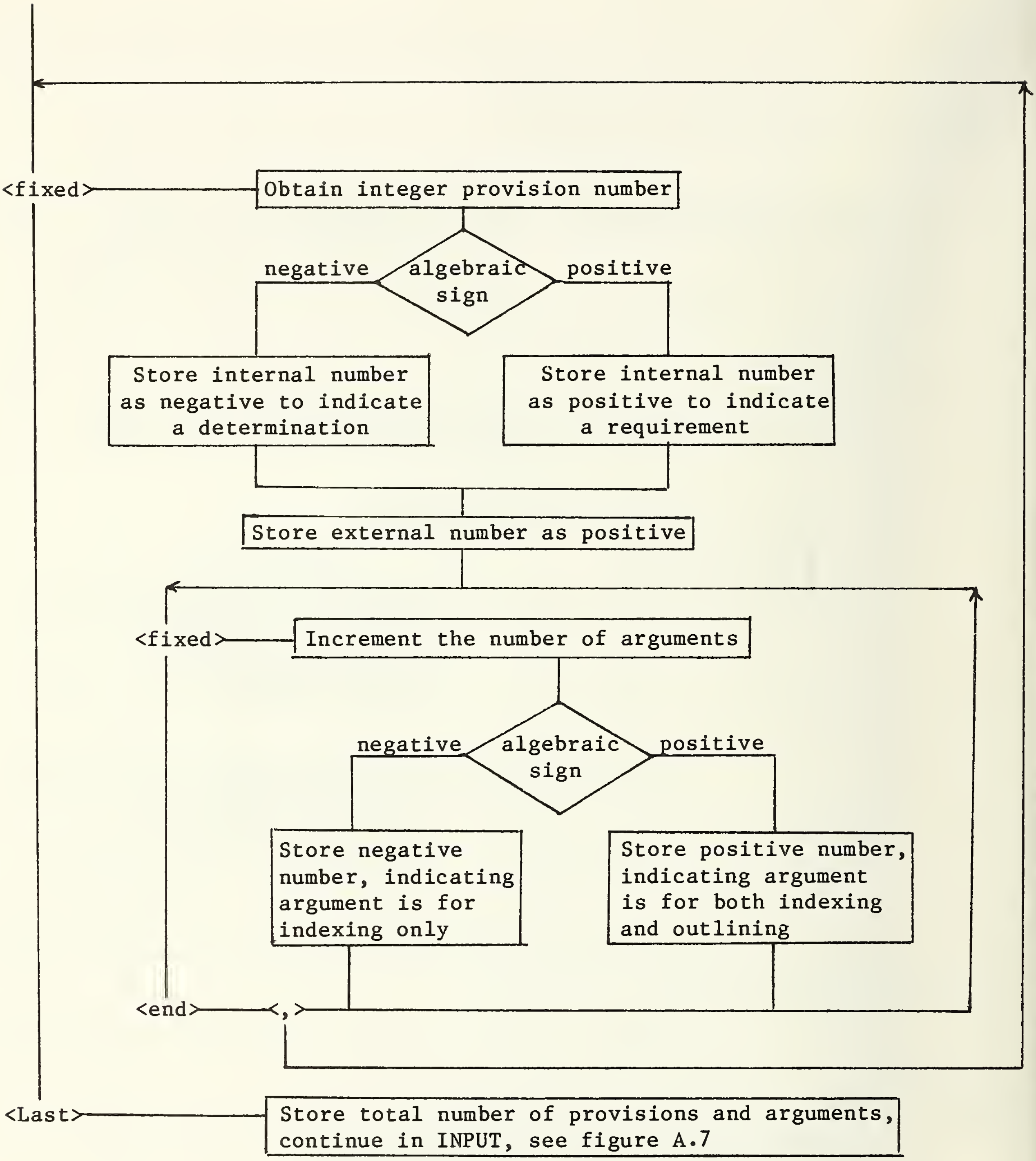

<fixed>
Obtain classifier number
Algebraic sign
negative
Mark the classifier as a "non-universial" root of a field for use in checking the full relevance criterion
positive
Increment the number of foster parents for the classifier
<Fixed>
Store the foster parent
<+>
<fixed>
Store the classifier as context required to be present for relevant facet appending
<->
<fixed>
Store the classifier as context required to be absent for relevant facet appending
</>
<end>
<LAST>
Store the total number of foster parents and context associations and continue in INPUT, see figure A.7

Figure A.9 Entry of Secondary Hierarchy Among Classifiers (INPUT)



<fixed>
Obtain integer provision number
negative
algebraic sign
positive
Store internal number as negative to indicate a determination
Store internal number as positive to indicate a requirement
Store external number as positive
<fixed>
Increment the number of arguments
negative
algebraic sign
positive
Store negative number, indicating argument is for indexing only
Store positive number, indicating argument is for both indexing and outlining
<end>
<,>
<Last>
Store total number of provisions and arguments, continue in INPUT, see figure A.7

Figure A.10 Entry of Provisions and Their Arguments (INPUT)

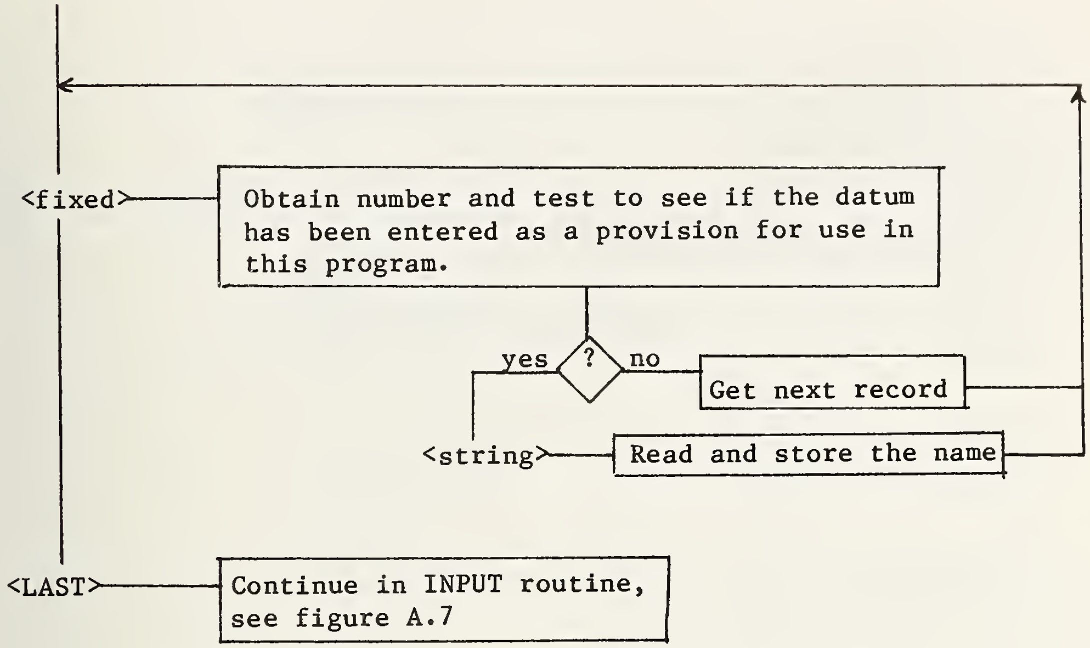
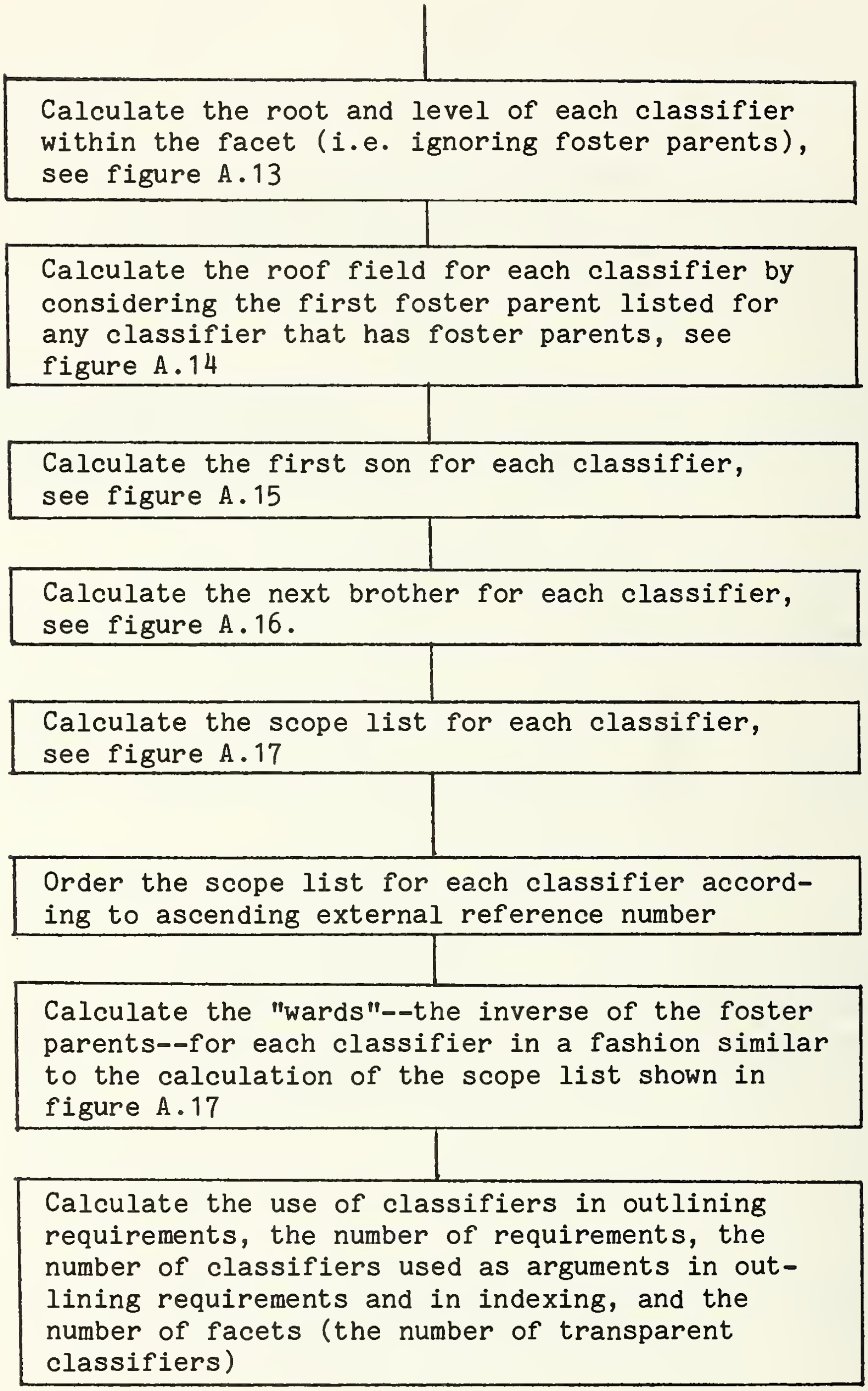

<fixed>
Obtain number and test to see if the datum has been entered as a provision for use in this program.
yes
?
no
Get next record
<string>
Read and store the name
<LAST>
Continue in INPUT routine, see figure A.7

Figure A.11 Entry of Provision Titles (INPUT)



Calculate the root and level of each classifier within the facet (i.e. ignoring foster parents), see figure A.13

Calculate the roof field for each classifier by considering the first foster parent listed for any classifier that has foster parents, see figure A.14

Calculate the first son for each classifier, see figure A.15

Calculate the next brother for each classifier, see figure A.16.

Calculate the scope list for each classifier, see figure A.17

Order the scope list for each classifier according to ascending external reference number

Calculate the "wards"--the inverse of the foster parents--for each classifier in a fashion similar to the calculation of the scope list shown in figure A.17

Calculate the use of classifiers in outlining requirements, the number of requirements, the number of classifiers used as arguments in outlining requirements and in indexing, and the number of facets (the number of transparent classifiers)

Figure A.12 Sequence for Preliminary Analysis (TOPANL)

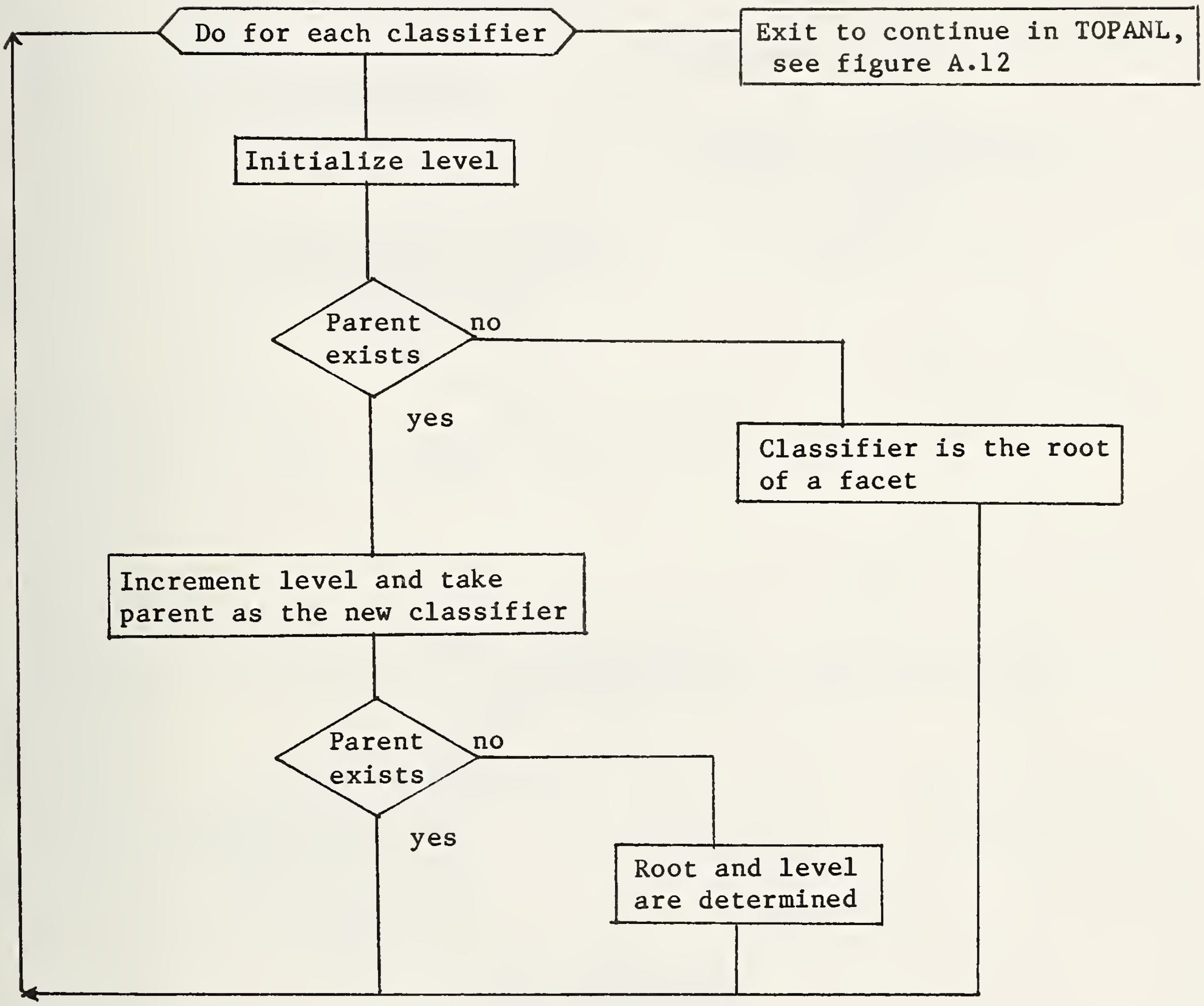

Do for each classifier
Exit to continue in TOPANL, see figure A.12
Initialize level
Parent exists
no
yes
Classifier is the root of a facet
Increment level and take parent as the new classifier
Parent exists
no
yes
Root and level are determined

Figure A.13 Calculation of Root and Level within a Facet (TOPANL)

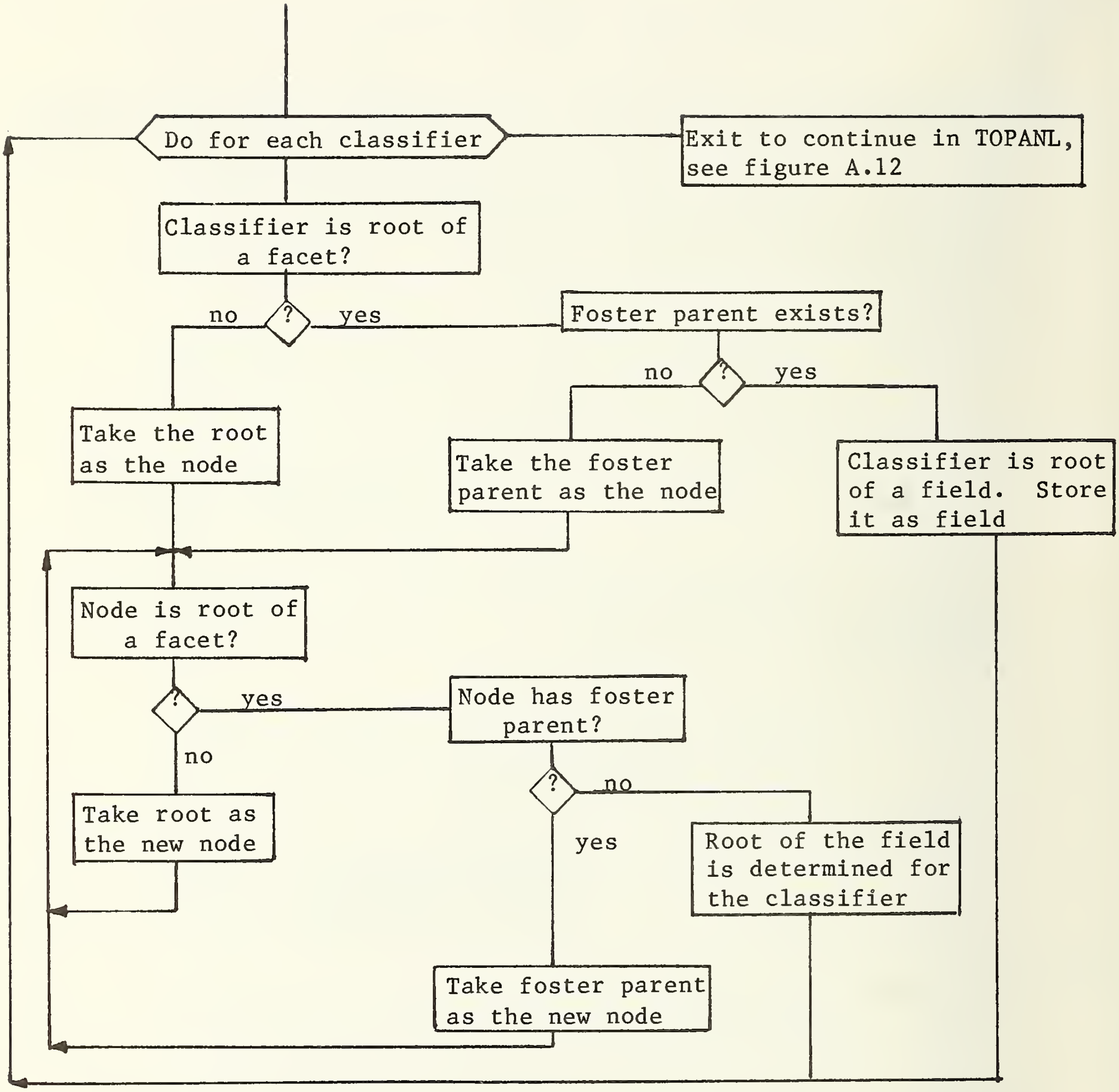
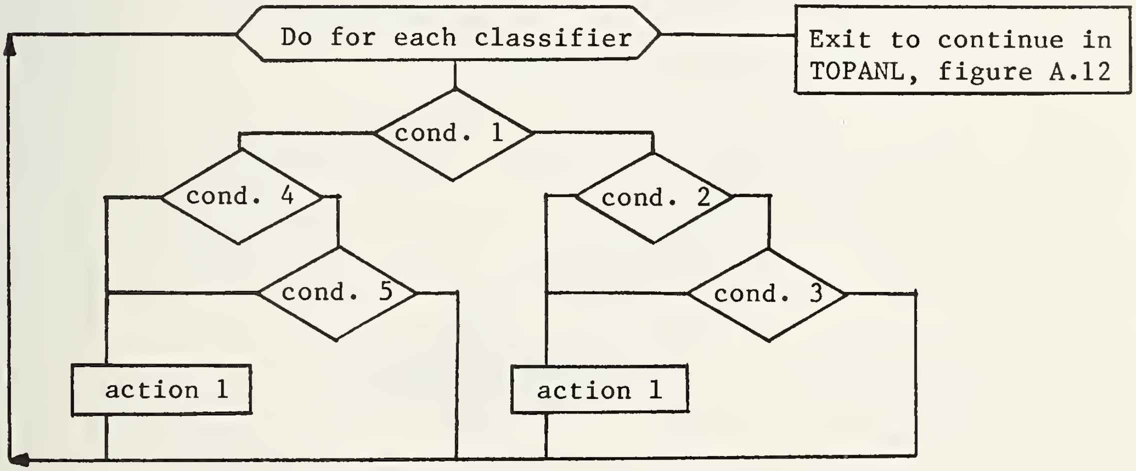

Do for each classifier
Exit to continue in TOPANL, see figure A.12
Classifier is root of a facet?
no
?
yes
Foster parent exists?
no
?
yes
Take the root as the node
Take the foster parent as the node
Classifier is root of a field. Store it as field
Node is root of a facet?
?
yes
Node has foster parent?
no
Take root as the new node
?
no
yes
Root of the field is determined for the classifier
Take foster parent as the new node

Figure A.14 Calculation of the Root to a Field (TOPANL)

		1	2	3	4	5	6
1	Classifier has parent	Y	Y	Y	N	N	N
2	Parent already has a first son	N	Y	Y	.	.	.
3	Parent's first son precedes this classifier	.	N	Y	.	.	.
4	Ficticious parent of all roots has a first son	.	.	.	N	Y	Y
5	Son of ficticious parent precedes this classifier	.	.	.	.	N	Y
1	Record classifier as the first	X	X		X	X	
2	Proceed to next classifier and repeat	X	X	X	X	X	X

a) decision table

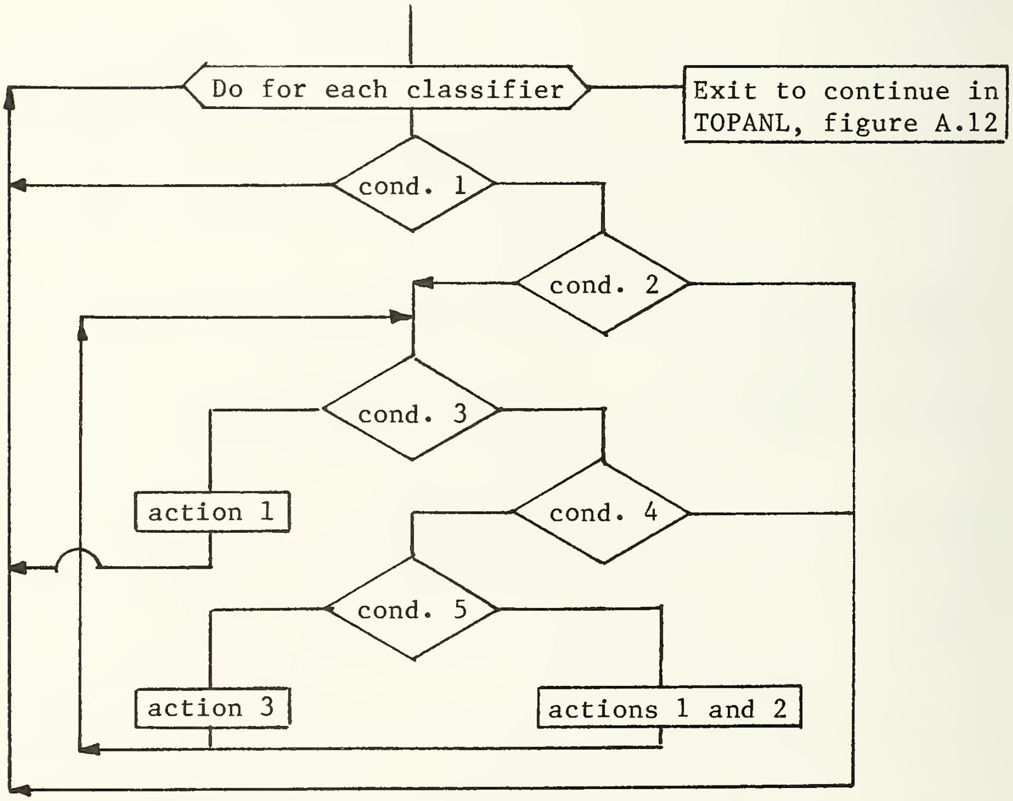


b) decision tree

Figure A.15 Calculation of First Son for a Classifier (TOPANL)

1	Classifier has parent	N	Y	Y	Y	Y	Y
2	Classifier is son of parent	.	Y	N	N	N	N
3	Brother of son exists	.	.	N	Y	Y	Y
4	Classifier is the brother	.	.	.	Y	N	N
5	Classifier precedes the brother	.	.	.	.	Y	N
1	Record classifier as the brother of the son			X		X	
2	Take classifier as new son and old brother as new classifier					X	
3	Take brother as new son						X
4	Reenter this table at condition 3					X	X
5	Proceed to next classifier and repeat	X	X	X	X		

a) decision table



b) decision tree

Figure A.16 Calculation of the Next Brother of a Classifier (TOPANL)

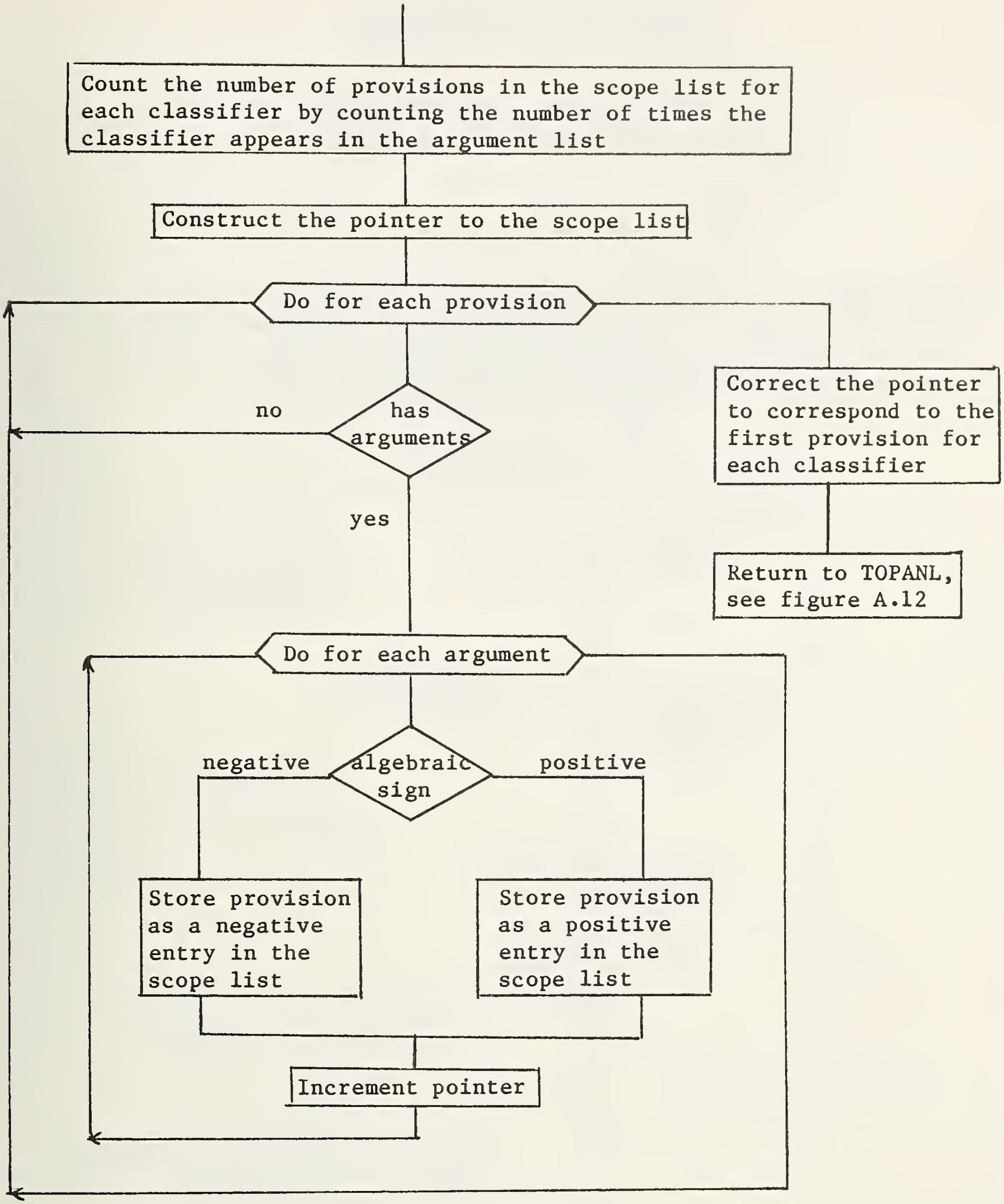

Count the number of provisions in the scope list for each classifier by counting the number of times the classifier appears in the argument list
Construct the pointer to the scope list
Do for each provision
no
has arguments
yes
Correct the pointer to correspond to the first provision for each classifier
Return to TOPANL, see figure A.12
Do for each argument
negative
algebraic sign
positive
Store provision as a negative entry in the scope list
Store provision as a positive entry in the scope list
Increment pointer

Figure A.17 Calculation of Scope List for a Classifier (TOPO)

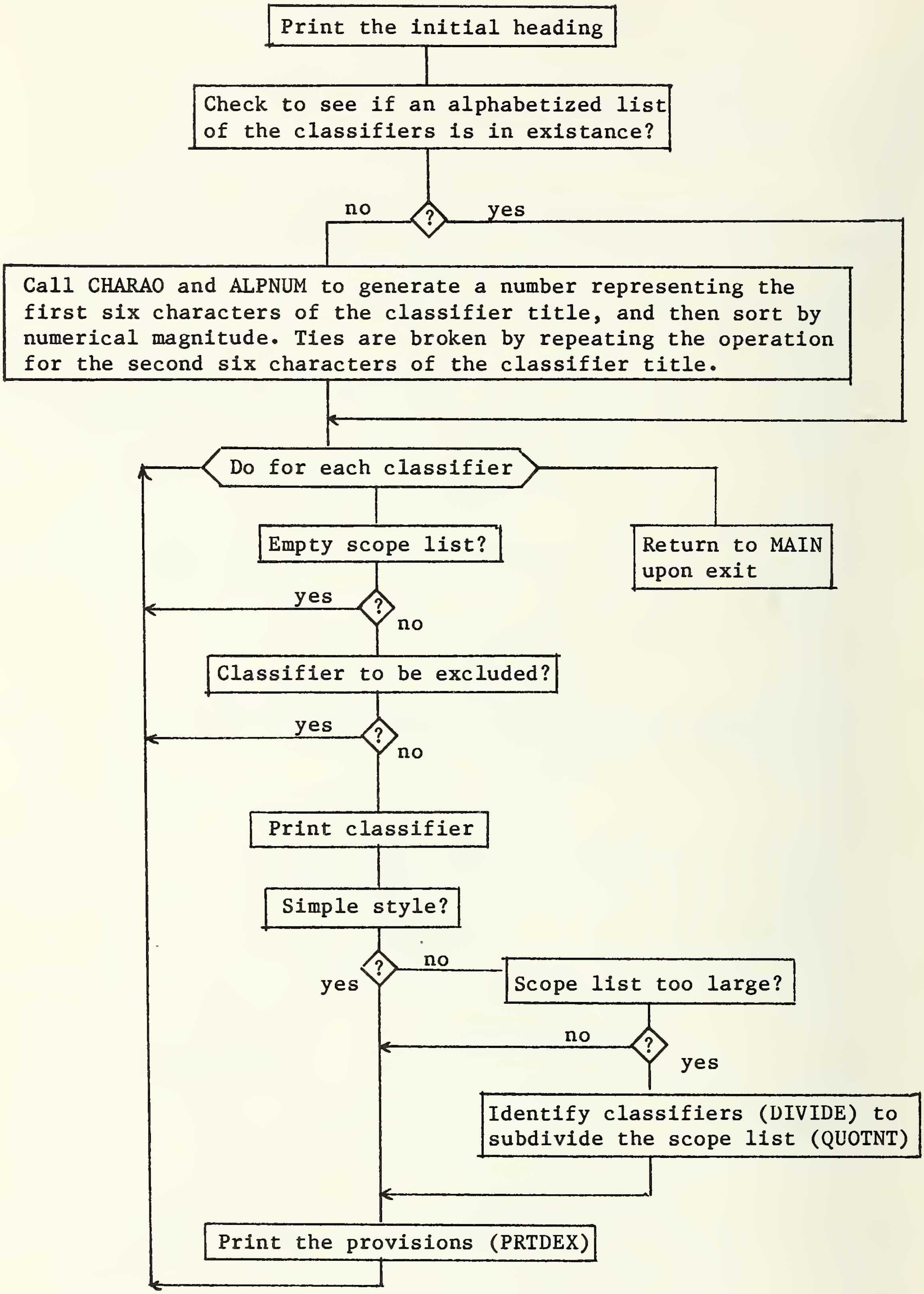

Print the initial heading
Check to see if an alphabetized list of the classifiers is in existance?
?
no
yes
Call CHARAO and ALPNUM to generate a number representing the first six characters of the classifier title, and then sort by numerical magnitude. Ties are broken by repeating the operation for the second six characters of the classifier title.
Do for each classifier
Return to MAIN upon exit
Empty scope list?
?
yes
no
Classifier to be excluded?
?
yes
no
Print classifier
Simple style?
?
no
yes
Scope list too large?
?
no
yes
Identify classifiers (DIVIDE) to subdivide the scope list (QUOTNT)
Print the provisions (PRTDEX)

Figure A.18 INDEX Generation

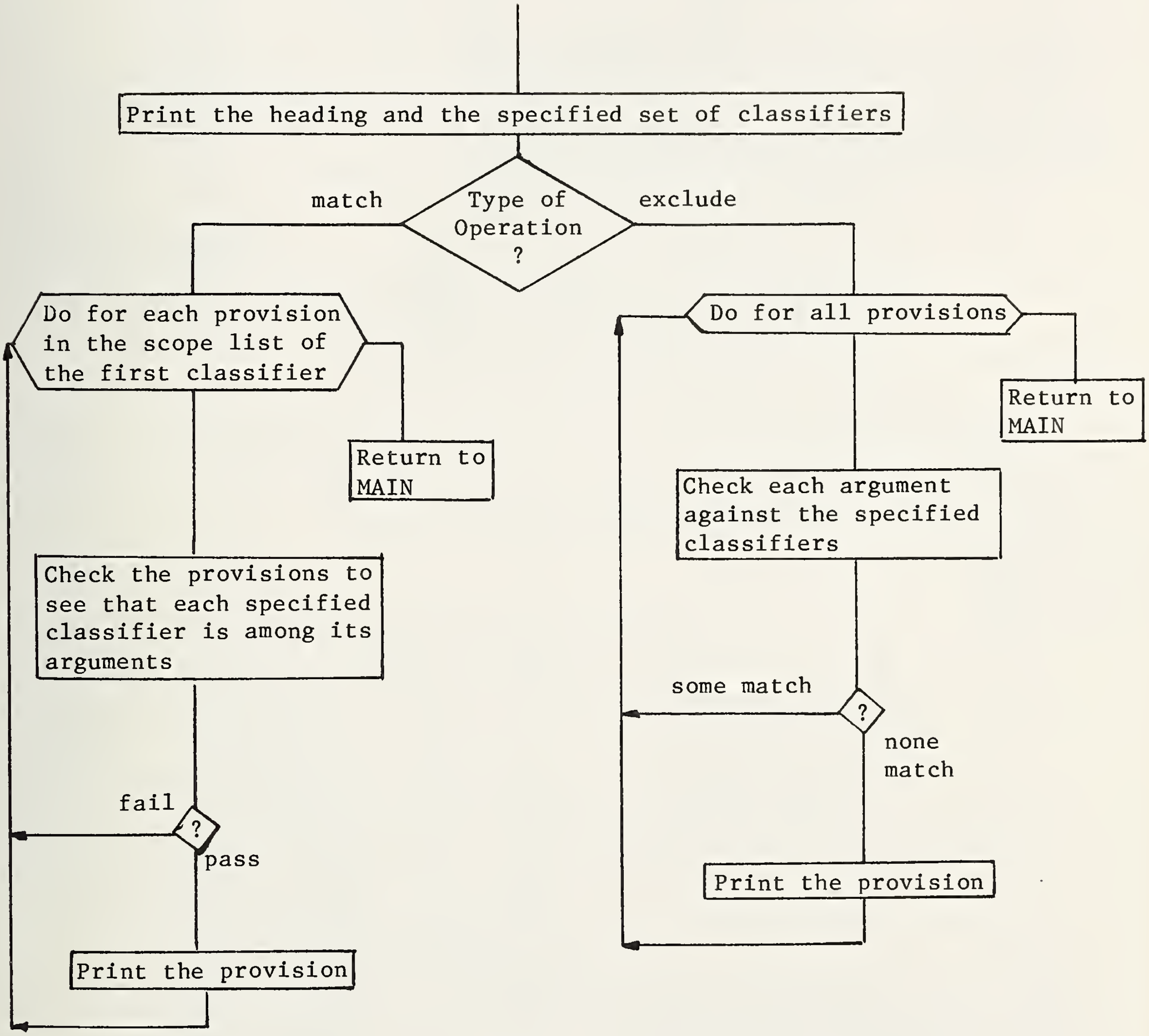

Print the heading and the specified set of classifiers
match
Type of Operation ?
exclude
Do for each provision in the scope list of the first classifier
Return to MAIN
Check the provisions to see that each specified classifier is among its arguments
fail
?
pass
Print the provision
Do for all provisions
Return to MAIN
Check each argument against the specified classifiers
some match
?
none match
Print the provision

Figure A.19 Indentification of Provisions for Specified Classifiers (SORT)

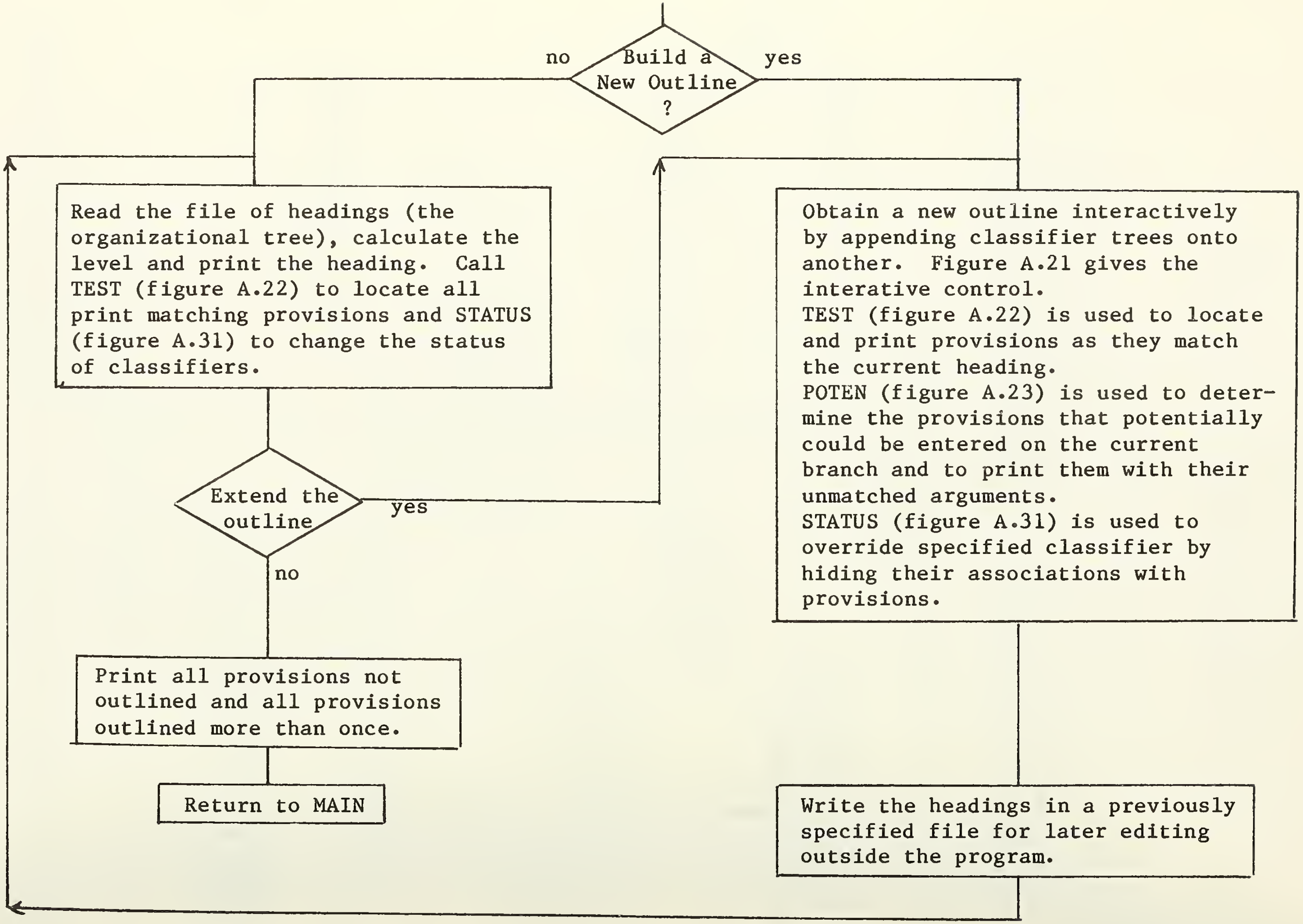

Build a New Outline ?
no
yes
Read the file of headings (the organizational tree), calculate the level and print the heading. Call TEST (figure A.22) to locate all print matching provisions and STATUS (figure A.31) to change the status of classifiers.
Extend the outline
yes
no
Print all provisions not outlined and all provisions outlined more than once.
Return to MAIN
Obtain a new outline interactively by appending classifier trees onto another. Figure A.21 gives the interative control.
TEST (figure A.22) is used to locate and print provisions as they match the current heading.
POTEN (figure A.23) is used to determine the provisions that potentially could be entered on the current branch and to print them with their unmatched arguments.
STATUS (figure A.31) is used to override specified classifier by hiding their associations with provisions.
Write the headings in a previously specified file for later editing outside the program.

Figure A.20 Flow of Control for Outlining (OUTLIN)

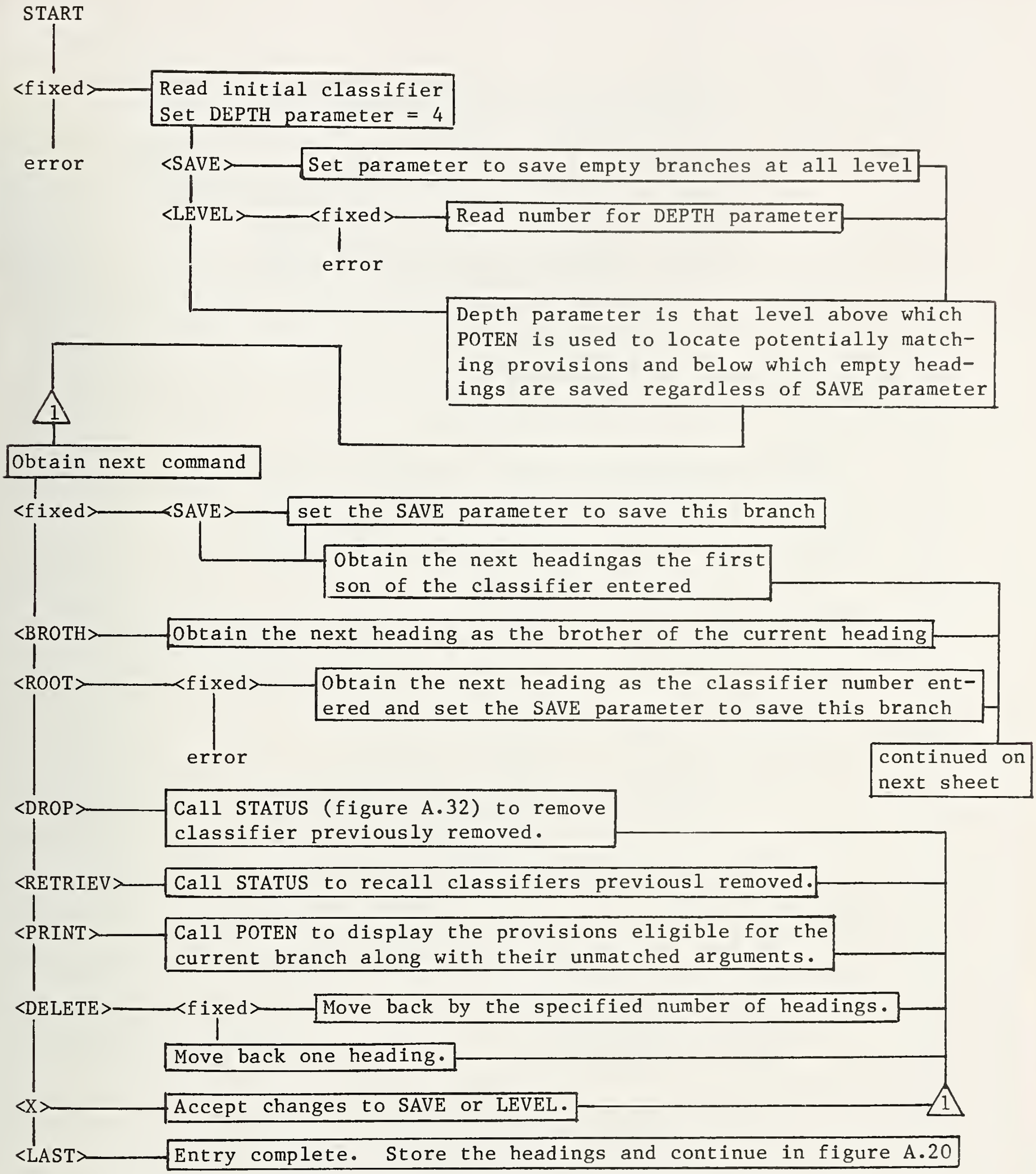

START
<fixed>
Read initial classifier
Set DEPTH parameter = 4
error
<SAVE>
Set parameter to save empty branches at all level
<LEVEL>
<fixed>
Read number for DEPTH parameter
error
Depth parameter is that level above which POTEN is used to locate potentially matching provisions and below which empty headings are saved regardless of SAVE parameter
1
Obtain next command
<fixed>
<SAVE>
set the SAVE parameter to save this branch
Obtain the next headingas the first son of the classifier entered
<BROTH>
Obtain the next heading as the brother of the current heading
<ROOT>
<fixed>
Obtain the next heading as the classifier number entered and set the SAVE parameter to save this branch
error
continued on next sheet
<DROP>
Call STATUS (figure A.32) to remove classifier previously removed.
<RETRIEV>
Call STATUS to recall classifiers previousl removed.
<PRINT>
Call POTEN to display the provisions eligible for the current branch along with their unmatched arguments.
<DELETE>
<fixed>
Move back by the specified number of headings.
Move back one heading.
<X>
Accept changes to SAVE or LEVEL.
1
<LAST>
Entry complete. Store the headings and continue in figure A.20

Figure A.21 Interactive Language Grammar for Outlining (OUTLIN)
-- concluded on next sheet--

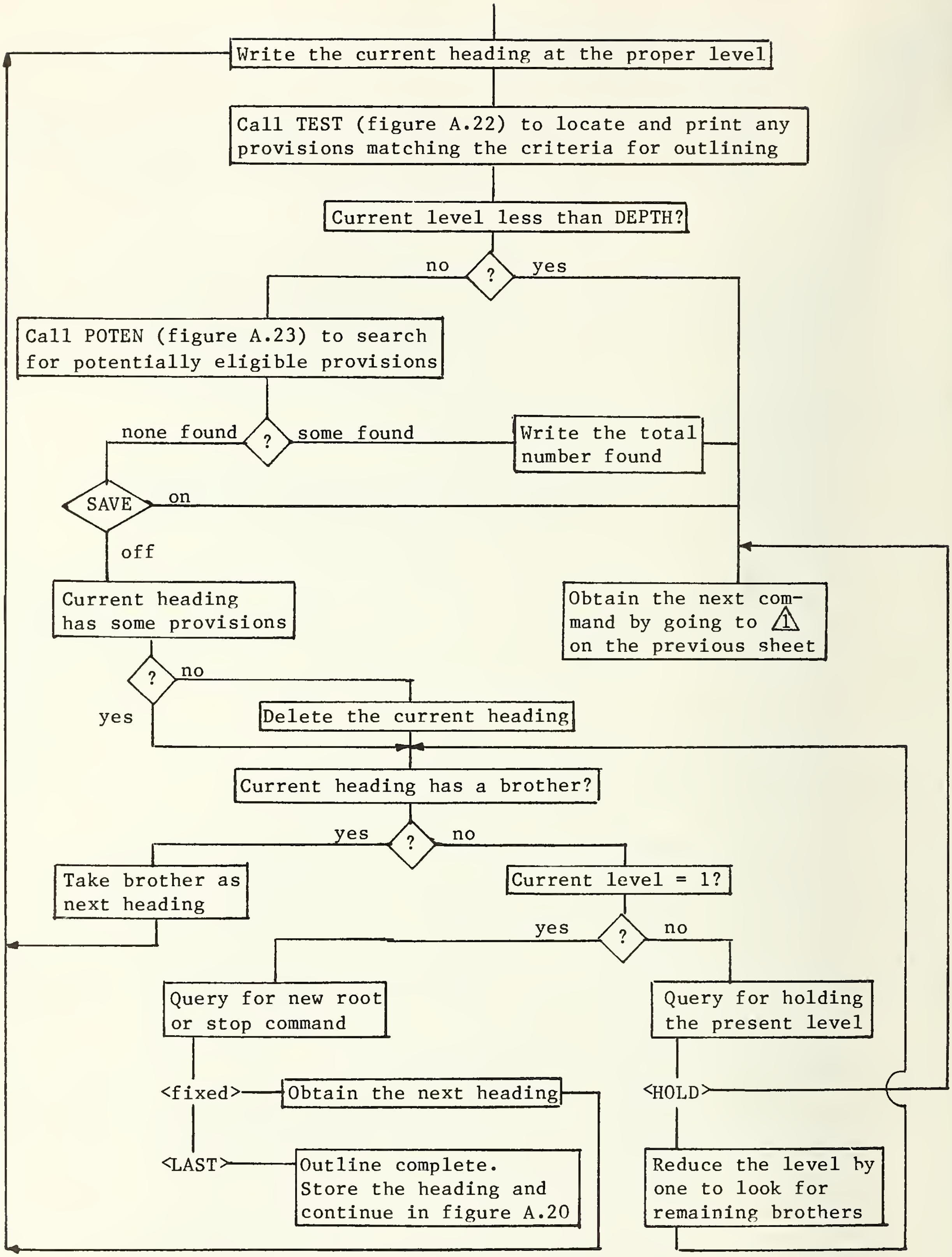

Write the current heading at the proper level
Call TEST (figure A.22) to locate and print any provisions matching the criteria for outlining
Current level less than DEPTH?
no
?
yes
Call POTEN (figure A.23) to search for potentially eligible provisions
none found
?
some found
Write the total number found
SAVE
on
off
Current heading has some provisions
Obtain the next command by going to △1 on the previous sheet
?
no
yes
Delete the current heading
Current heading has a brother?
yes
?
no
Take brother as next heading
Current level = 1?
yes
?
no
Query for new root or stop command
Query for holding the present level
<fixed>
Obtain the next heading
<HOLD>
<LAST>
Outline complete. Store the heading and continue in figure A.20
Reduce the level by one to look for remaining brothers

Figure A.21 Interactive Language Grammar for Outlining (OUTLIN)
-- continued from previous sheet --

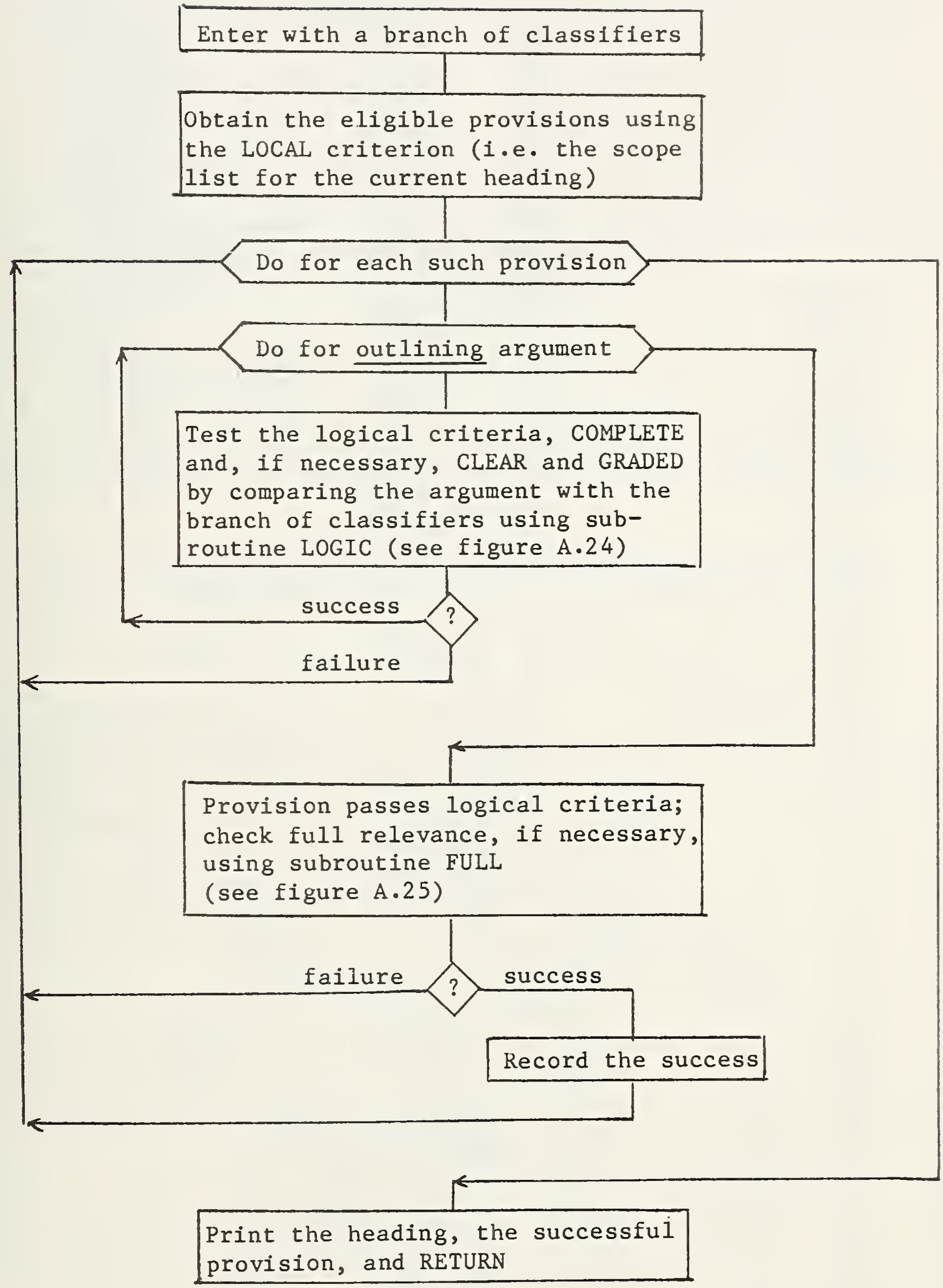

Enter with a branch of classifiers
Obtain the eligible provisions using the LOCAL criterion (i.e. the scope list for the current heading)
Do for each such provision
Do for outlining argument
Test the logical criteria, COMPLETE and, if necessary, CLEAR and GRADED by comparing the argument with the branch of classifiers using subroutine LOGIC (see figure A.24)
success
?
failure
Provision passes logical criteria; check full relevance, if necessary, using subroutine FULL (see figure A.25)
failure
?
success
Record the success
Print the heading, the successful provision, and RETURN

Figure A.22 Identification of Provisions to be Outlined (TEST)

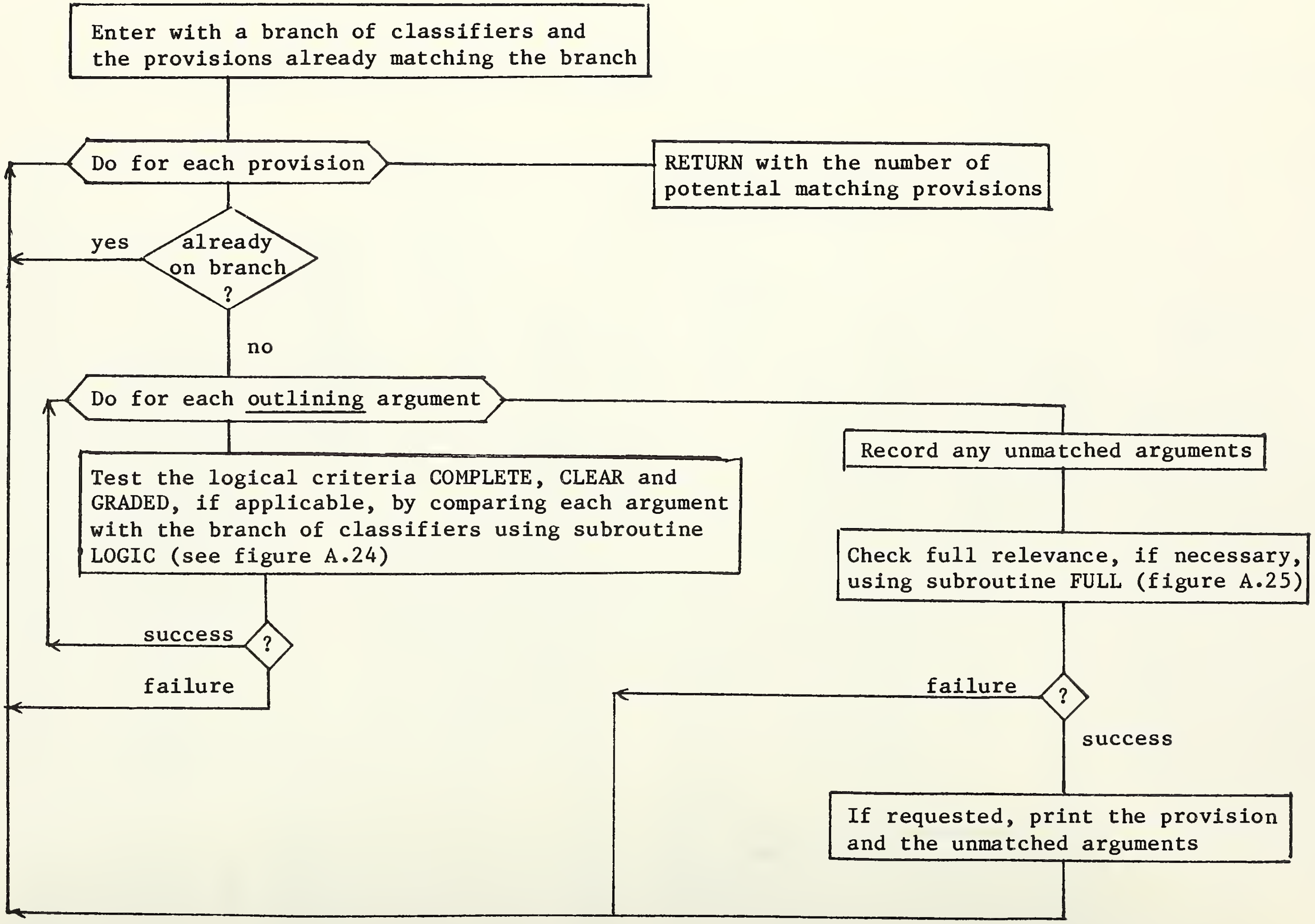
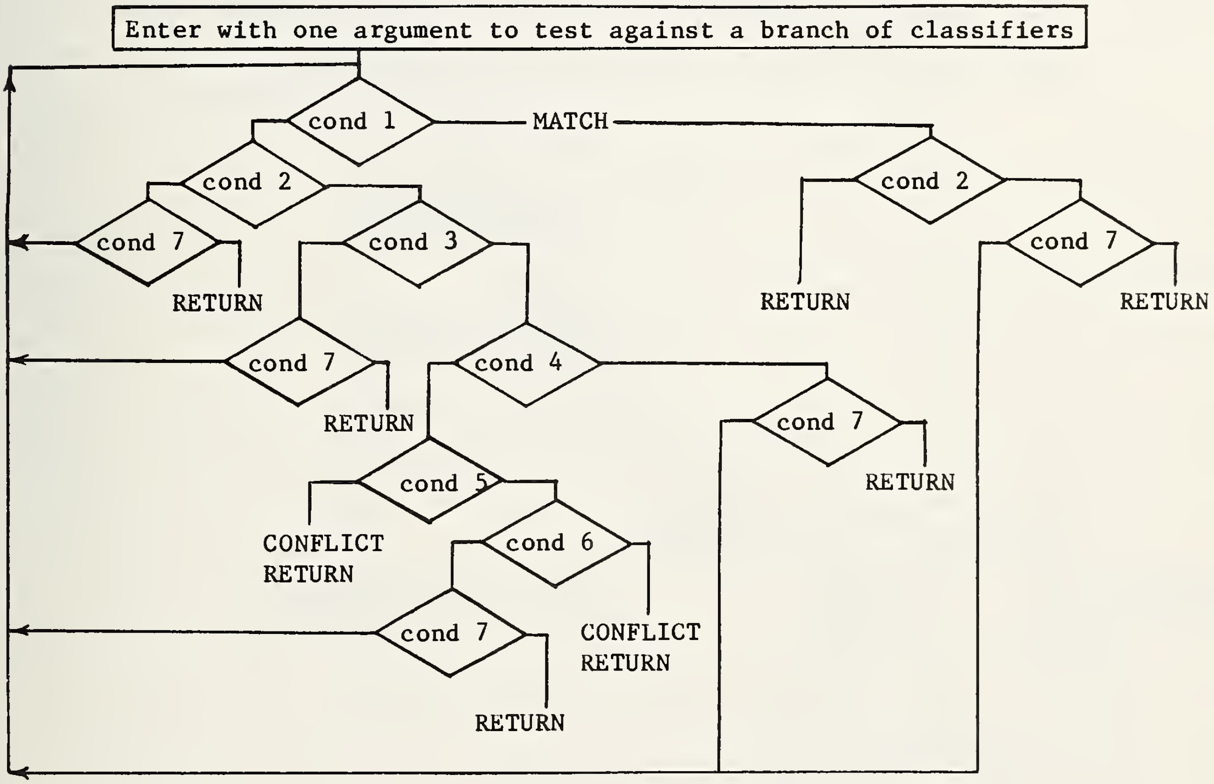

Enter with a branch of classifiers and the provisions already matching the branch
Do for each provision
RETURN with the number of potential matching provisions
yes
already on branch ?
no
Do for each outlining argument
Test the logical criteria COMPLETE, CLEAR and GRADED, if applicable, by comparing each argument with the branch of classifiers using subroutine LOGIC (see figure A.24)
success
?
failure
Record any unmatched arguments
Check full relevance, if necessary, using subroutine FULL (figure A.25)
failure
?
success
If requested, print the provision and the unmatched arguments

Figure A.23 Identification of Provisions Waiting to be Outlined (POTEN)

		1	2	3	4	5	6	7	8	9	10	11	12	13
1	Argument is the classifier	Y	Y	Y	N	N	N	N	N	N	N	N	N	N
2	UNIQUE criterion is in effect	N	Y	Y	N	N	Y	Y	Y	Y	Y	Y	Y	Y
3	Argument and classifier are in same facet	.	.	.	.	.	N	N	Y	Y	Y	Y	Y	Y
4	Classifier is a predecessor of the argument	.	.	.	.	.	.	.	Y	Y	N	N	N	N
5	Argument is a predecessor of the classifier	.	.	.	.	.	.	.	.	.	Y	Y	Y	N
6	GRADED criterion is in effect	.	.	.	.	.	.	.	.	.	Y	N	N	.
7	All classifiers checked	.	N	Y	N	Y	N	Y	N	Y	.	N	Y	.
1	Record a match	X	X	X										
2	Record a conflict										X			X
3	RETURN	X		X		X		X		X	X		X	X
4	Get next classifier and re-enter the table		X		X		X		X			X		

Note: The returns from rules 5, 7, 9, and 12 may indicate that the argument is unmatched, meaning the provision could still be outlined further along the branch

a) decision table



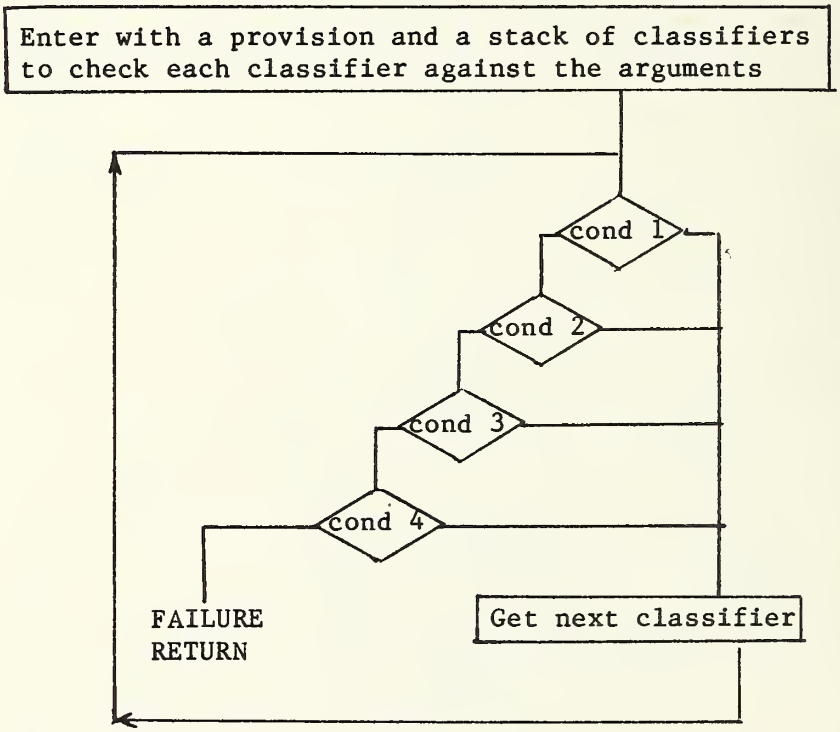
b) decision tree

Note: Conditions 4 and 5 are not single tests; the lower portion of figure A.26 shows the logic for a test very similar to condition 4.

Figure A.24 Checking Arguments for Logical Criteria (LOGIC)

		1	2	3	4	5
1	Classifier is universal root	Y	N	N	N	N
2	Classifier is dropped from consideration	.	Y	N	N	N
3	Classifier satisfies DIRECT Test (figure A.26)	.	.	Y	N	N
4	Classifier satisfies INDIRECT Test (figure A.27)	.	.	.	Y	N
1	Classifier is relevant, get the next one	X	X	X	X	
2	Provision fails					X

a) decision table


Enter with a provision and a stack of classifiers to check each classifier against the arguments
cond 1
cond 2
cond 3
cond 4
FAILURE RETURN
Get next classifier

b) decision tree

Figure A.25 Checking Provisions for Full Relevance (FULL)

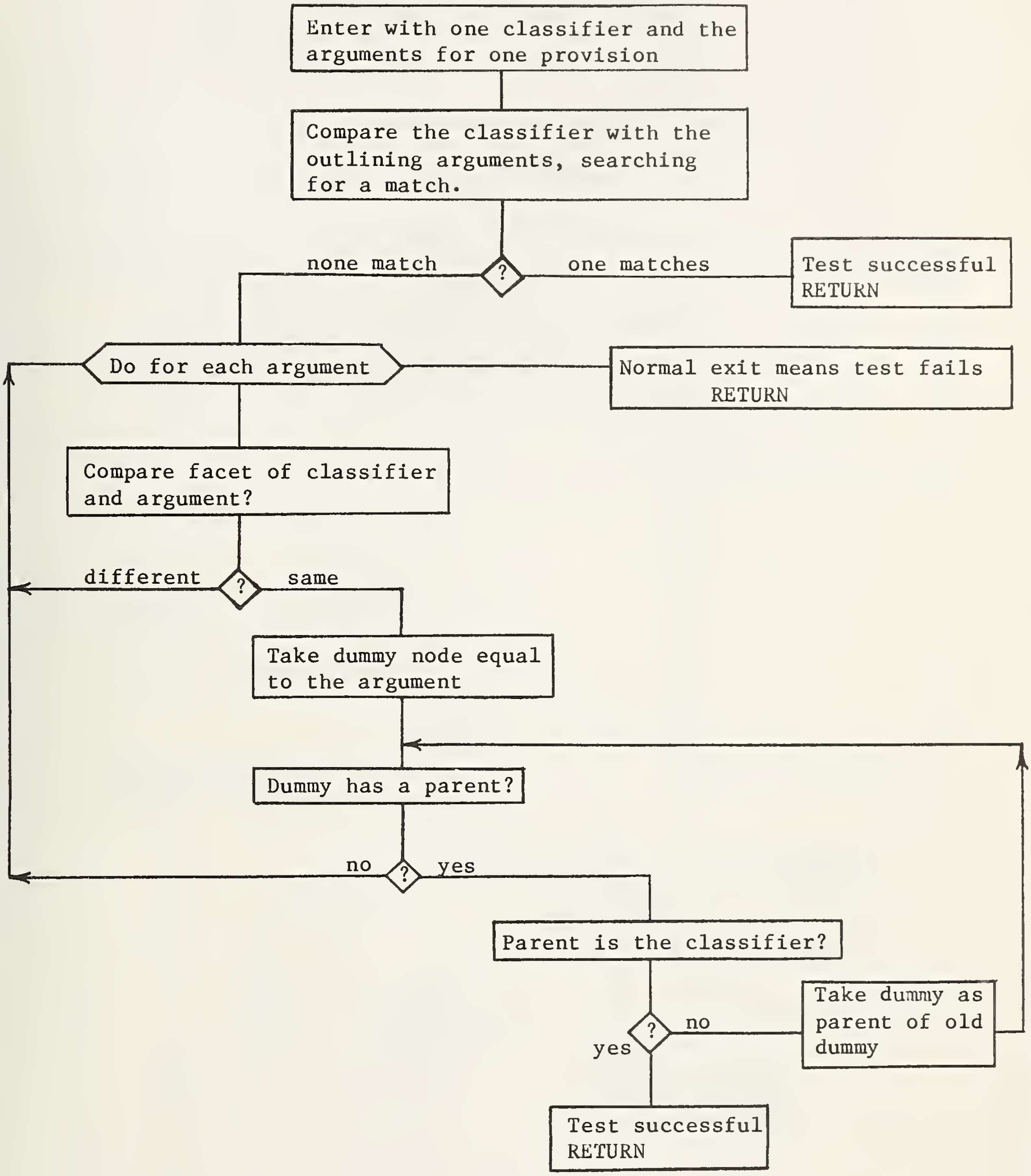

Enter with one classifier and the arguments for one provision
Compare the classifier with the outlining arguments, searching for a match.
none match
?
one matches
Test successful RETURN
Do for each argument
Normal exit means test fails RETURN
Compare facet of classifier and argument?
different
?
same
Take dummy node equal to the argument
Dummy has a parent?
no
?
yes
Parent is the classifier?
Take dummy as parent of old dummy
?
no
yes
Test successful RETURN

Figure A.26 Checking Classifier for Direct Relevance (DIRECT)

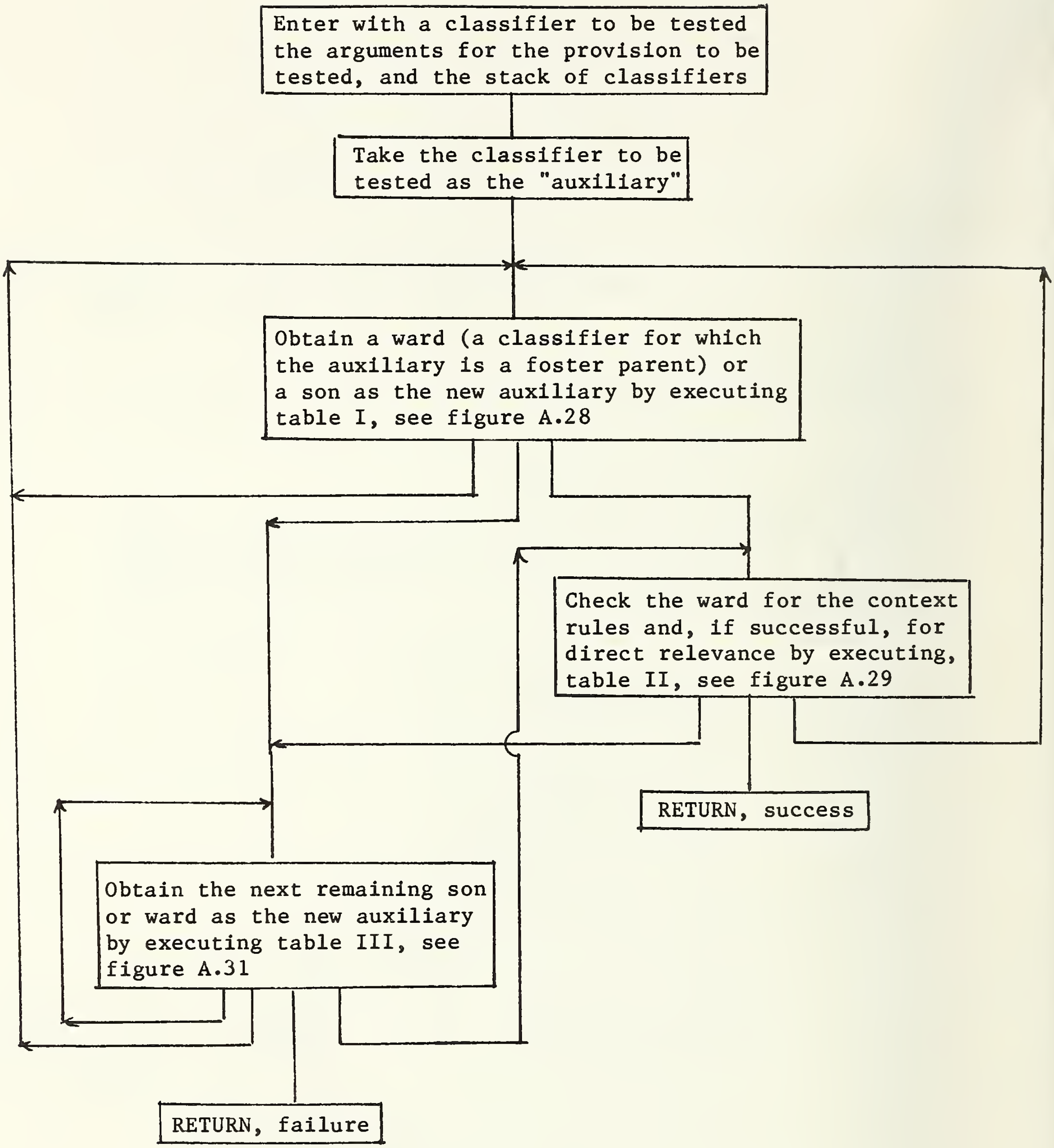
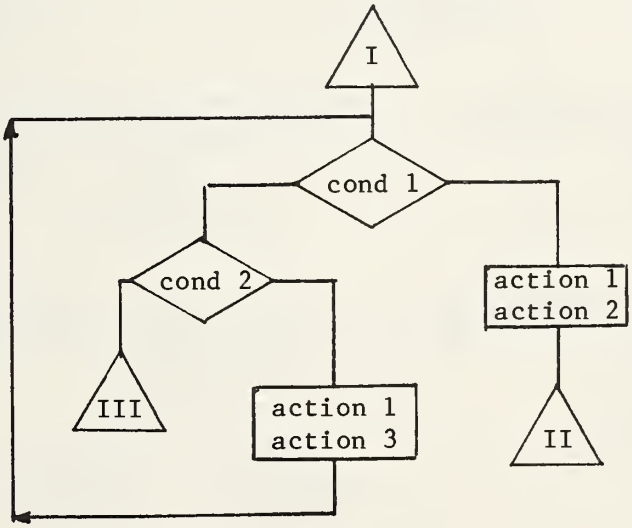

Enter with a classifier to be tested the arguments for the provision to be tested, and the stack of classifiers
Take the classifier to be tested as the "auxiliary"
Obtain a ward (a classifier for which the auxiliary is a foster parent) or a son as the new auxiliary by executing table I, see figure A.28
Check the ward for the context rules and, if successful, for direct relevance by executing, table II, see figure A.29
RETURN, success
Obtain the next remaining son or ward as the new auxiliary by executing table III, see figure A.31
RETURN, failure

Figure A.27 Overall Flow for Checking Indirect Relevance (INDRCT)

		1	2	3
1	Current auxiliary has a ward	Y	N	N
2	Current auxiliary has a son	.	Y	N
1	Increment the stack of auxiliaries	X	X	
2	Take the ward as the new auxiliary and record the number of wards remaining for the auxiliary at the prior level of the stack	X		
3	Take the son as the current auxiliary and record its next brother in the stack		X	
4	Execute table I (this table) again		X	
5	Execute table II (figure A.29)	X		
6	Execute table III (figure A.31)			X

a) decision table I

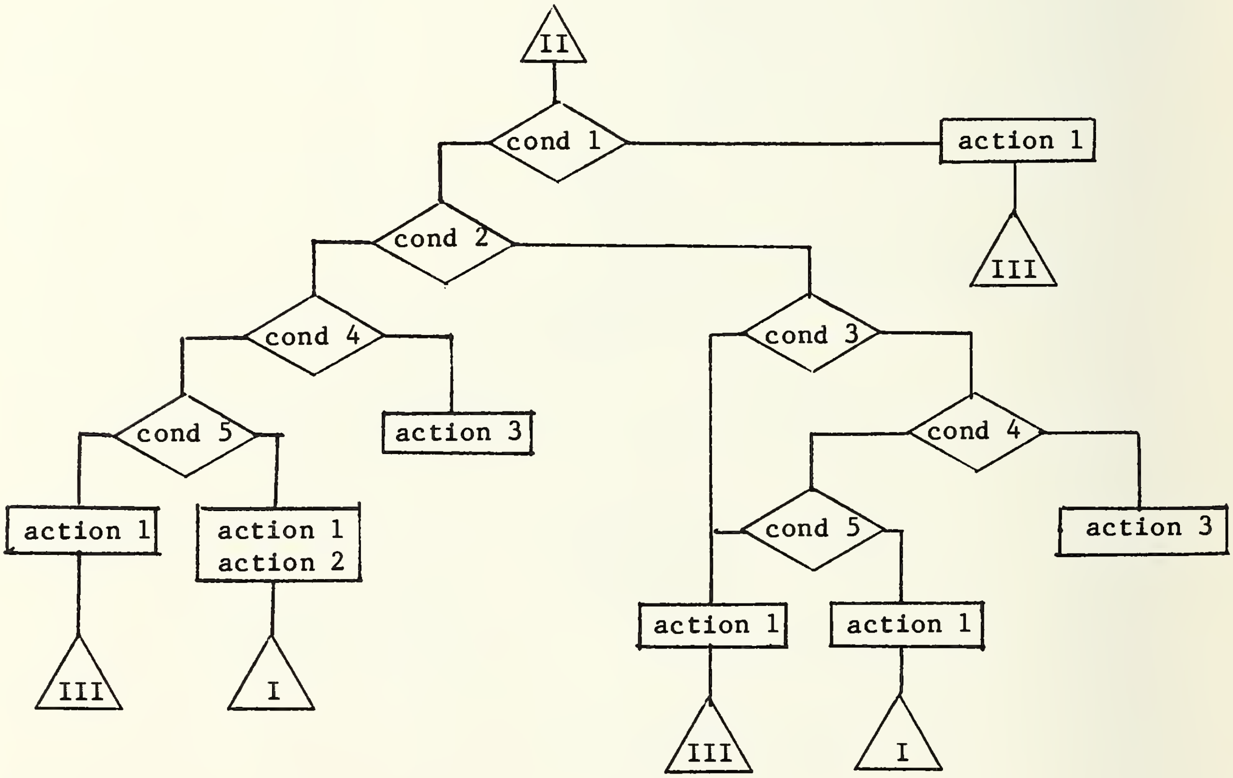


b) decision tree

Figure A.28 Identify Potential New Links in Secondary Hierarchy (INDRCT)

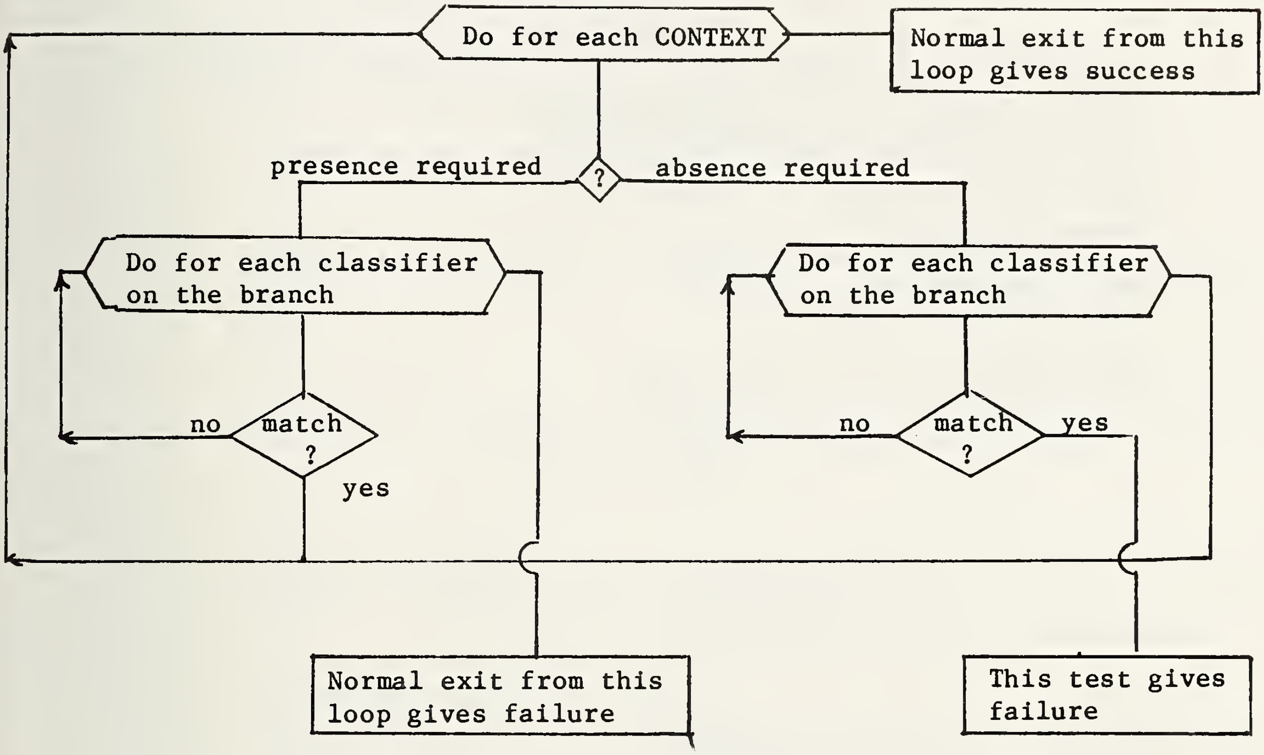
		1	2	3	4	5	6	7	8
1	Current auxiliary has already failed for this provision	Y	N	N	N	N	N	N	N
2	Current auxiliary has a context requirement	.	Y	Y	Y	Y	N	N	N
3	Context requirement is satisfied (see figure A.30)	.	N	Y	Y	Y	.	.	.
4	Current auxiliary satisfies DIRECT test (see figure A.26)	.	.	Y	N	N	Y	N	N
5	Current auxiliary has a son	.	.	.	Y	N	.	Y	N
1	Record a failure for this auxiliary	X	X		X	X		X	X
2	Increment the stack of auxiliaries, take the sone as the new auxiliary, and record its brother in the stack				X			X	
3	Test is successful, RETURN			X			X		
4	Execute table I (figure A.28)				X			X	
5	Execute table III (figure A.31)	X	X			X			X

a) decision table II



b) decision tree

Figure A.29 Checking the Secondary Hierarchy (INDRCT)

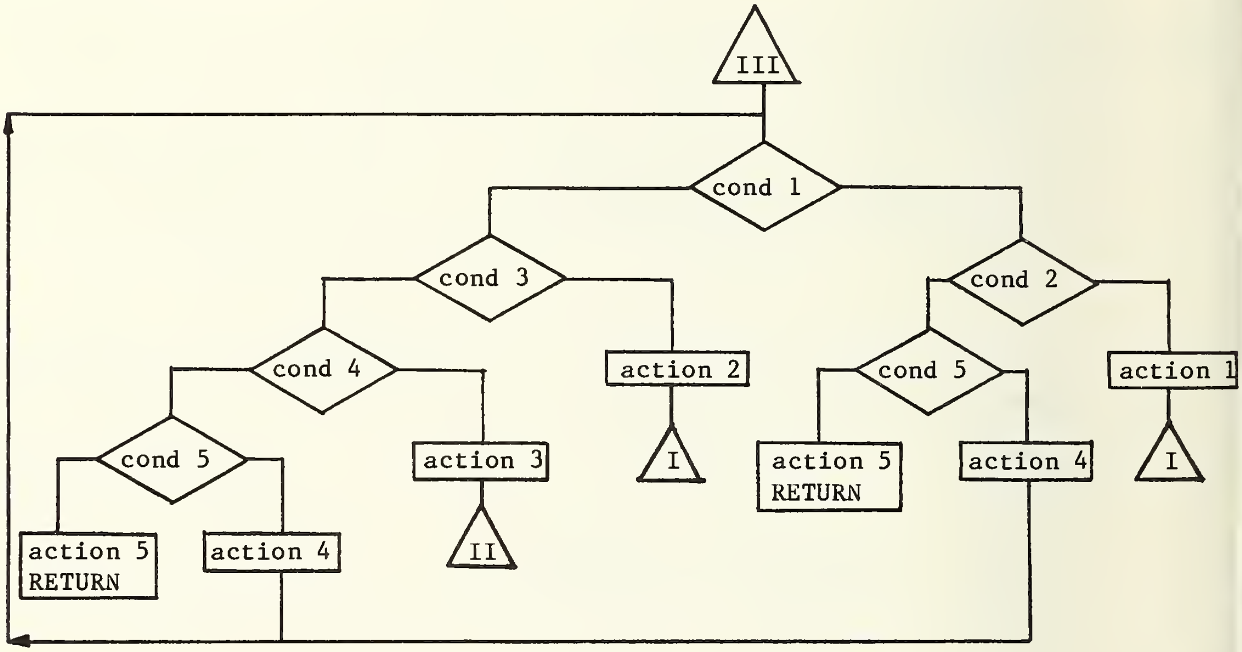

Do for each CONTEXT
Normal exit from this loop gives success
presence required
?
absence required
Do for each classifier on the branch
no
match ?
yes
Do for each classifier on the branch
no
match ?
yes
Normal exit from this loop gives failure
This test gives failure

Note: This corresponds to condition 4 of the decision table of figure A.29

Figure A.30 Check Context for the Secondary Hierarchy (INDRCT)

		1	2	3	4	5	6	7
1	Current auxiliary is son of the prior auxiliary (if not, it is a ward of the prior auxiliary)	Y	Y	Y	N	N	N	N
2	Next brother of current auxiliary remains	Y	N	N	.	.	.	.
3	Another ward of prior auxiliary remains	.	.	.	Y	N	N	N
4	Son of prior auxiliary exists	.	.	.	.	Y	N	N
5	Stack of auxiliaries is empty	.	Y	N	.	.	Y	N
1	Take brother a new auxiliary and record its brother for the stack	X						
2	Take next ward as new auxiliary and decrement the number of wards remaining for the prior auxiliary				X			
3	Take son as the new auxiliary and record its brother for the stack					X		
4	Decrement the stack of auxiliaries		X				X	
5	Test fails, RETURN			X				X
6	Execute table I (figure A.28)	X				X		
7	Execute table II (figure A.29)				X			
8	Execute table III (this table) again		X				X	

a) decision table III



b) decision tree

Figure A.31 Identify Remaining Links in Secondary Hierarchy (INDRCT)

]

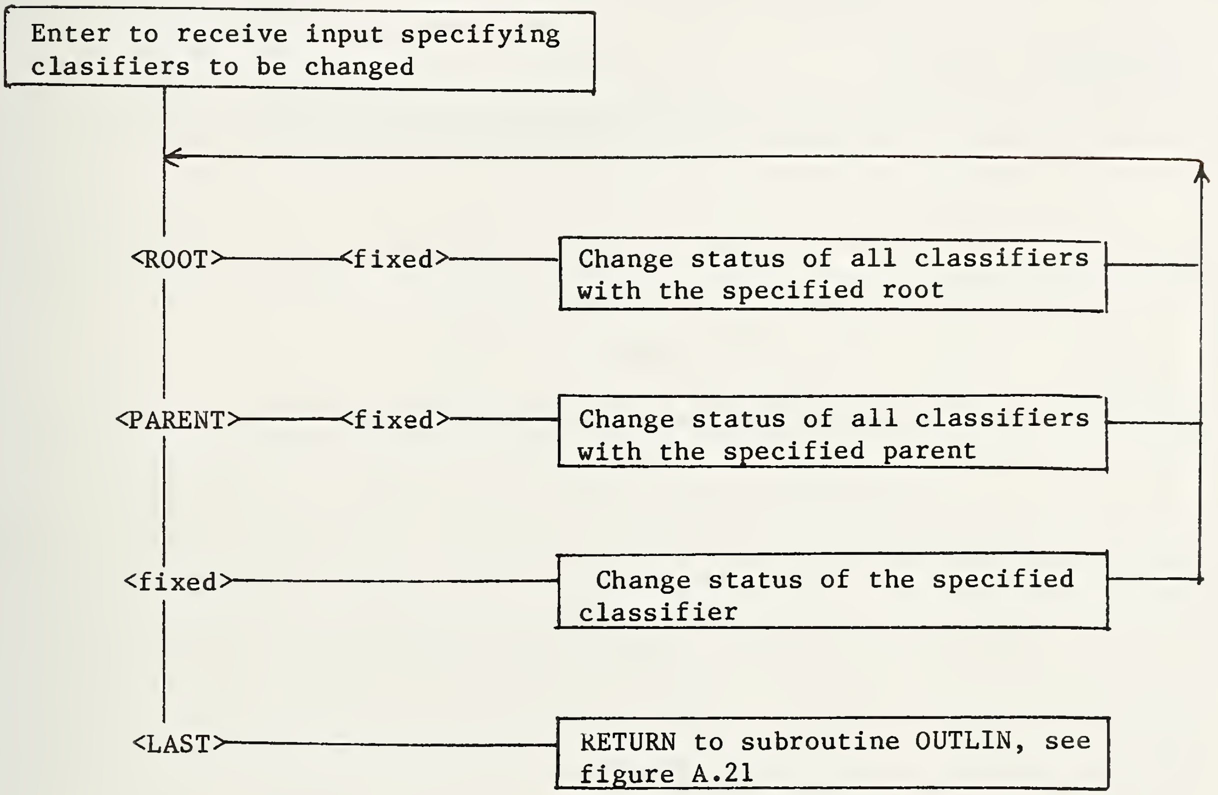

Enter to receive input specifying clasifiers to be changed
<ROOT>
<fixed>
Change status of all classifiers with the specified root
<PARENT>
<fixed>
Change status of all classifiers with the specified parent
<fixed>
Change status of the specified classifier
<LAST>
RETURN to subroutine OUTLIN, see figure A.21

Figure A.32 Change the Outlining Status of a Classifier (STATUS)

NBS-114A (REV. 2-80)

U.S. DEPT. OF COMM. **BIBLIOGRAPHIC DATA SHEET** *(See instructions)*	1. PUBLICATION OR REPORT NO. NBS BSS 136	2. Performing Organ. Report No.	3. Publication Date September 1981

4. TITLE AND SUBTITLE

ORGANIZATION OF BUILDING STANDARDS:
SYSTEMATIC TECHNIQUES FOR SCOPE AND ARRANGEMENT

5. AUTHOR(S)

James Robert Harris and Richard N. Wright

6. PERFORMING ORGANIZATION *(If joint or other than NBS, see instructions)*

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20234

7. Contract/Grant No.

8. Type of Report & Period Covered

Final

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS *(Street, City, State, ZIP)*

Same as item 6.

10. SUPPLEMENTARY NOTES

Library of Congress Catalog Card Number: 81-600124

☒ Document describes a computer program; SF-185, FIPS Software Summary, is attached.

11. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here)*

Standards should be organized so that they provide reliable and quick access to the provisions of the standard. Organization is considered to deal with both the scope and the arrangement of the provisions of a standard. It is found to have objective qualities that allow it to be treated formally. Necessary and desirable qualities for an organization are identified, verified, and adopted as objectives and guidelines. The basic element of the system for organizing standards is the classification of the provisions of a standard. A faceted structure, providing a clear division between those levels that are strictly logical and those that are not, is recommended for the classification system. A relevant basis is found for classifying requirements using an idealized model of the relation between syntax and semantics. Development of the classification constitutes a formal treatment of scope. The classification is easily transformed into an index. Development of an outline from the classifiers constitutes a formal treatment of the arrangement. Criteria for placement of provisions in outlines and for construction of outlines from the classification are proposed to promote the objectives of organization. A computer algorithm for interactive outline generation is developed and evaluated. Measures are defined for the comparison of alternate outlines for the same standard.

12. KEY WORDS *(Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)*

arrangement; building; classification; code; engineering; organization; provisions; scope; specification; standard; system analysis/engineering.

13. AVAILABILITY

☒ Unlimited
☐ For Official Distribution. Do Not Release to NTIS
☒ Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D C 20402.
☐ Order From National Technical Information Service (NTIS), Springfield, VA. 22161

14. NO. OF PRINTED PAGES

278

15. Price

$7.00

☆U.S. GOVERNMENT PRINTING OFFICE: 1981-340-997/1824

NBS TECHNICAL PUBLICATIONS

PERIODICALS

JOURNAL OF RESEARCH—The Journal of Research of the National Bureau of Standards reports NBS research and development in those disciplines of the physical and engineering sciences in which the Bureau is active. These include physics, chemistry, engineering, mathematics, and computer sciences. Papers cover a broad range of subjects, with major emphasis on measurement methodology and the basic technology underlying standardization. Also included from time to time are survey articles on topics closely related to the Bureau's technical and scientific programs. As a special service to subscribers each issue contains complete citations to all recent Bureau publications in both NBS and non-NBS media. Issued six times a year. Annual subscription: domestic $13; foreign $16.25. Single copy, $3 domestic; $3.75 foreign.

NOTE: The Journal was formerly published in two sections: Section A "Physics and Chemistry" and Section B "Mathematical Sciences."

DIMENSIONS/NBS—This monthly magazine is published to inform scientists, engineers, business and industry leaders, teachers, students, and consumers of the latest advances in science and technology, with primary emphasis on work at NBS. The magazine highlights and reviews such issues as energy research, fire protection, building technology, metric conversion, pollution abatement, health and safety, and consumer product performance. In addition, it reports the results of Bureau programs in measurement standards and techniques, properties of matter and materials, engineering standards and services, instrumentation, and automatic data processing. Annual subscription: domestic $11; foreign $13.75.

NONPERIODICALS

Monographs—Major contributions to the technical literature on various subjects related to the Bureau's scientific and technical activities.

Handbooks—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications—Include proceedings of conferences sponsored by NBS, NBS annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

Applied Mathematics Series—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

National Standard Reference Data Series—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a worldwide program coordinated by NBS under the authority of the National Standard Data Act (Public Law 90-396).

NOTE: The principal publication outlet for the foregoing data is the Journal of Physical and Chemical Reference Data (JPCRD) published quarterly for NBS by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements available from ACS, 1155 Sixteenth St., NW, Washington, DC 20056.

Building Science Series—Disseminates technical information developed at the Bureau on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NBS under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The standards establish nationally recognized requirements for products, and provide all concerned interests with a basis for common understanding of the characteristics of the products. NBS administers this program as a supplement to the activities of the private sector standardizing organizations.

Consumer Information Series—Practical information, based on NBS research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

Order the **above** *NBS publications from: Superintendent of Documents, Government Printing Office, Washington, DC 20402.*

Order the **following** *NBS publications—FIPS and NBSIR's—from the National Technical Information Services, Springfield, VA 22161.*

Federal Information Processing Standards Publications (FIPS PUB)—Publications in this series collectively constitute the Federal Information Processing Standards Register. The Register serves as the official source of information in the Federal Government regarding standards issued by NBS pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

NBS Interagency Reports (NBSIR)—A special series of interim or final reports on work performed by NBS for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Services, Springfield, VA 22161, in paper copy or microfiche form.